ՀԱՅԿԱԿԱՆ ՍՍՀ ԳԻՏՈՒԹՅՈՒՆՆԵՐԻ ԱԿԱԴԵՄԻԱՅԻ ԶԵԿՈՒՅՑՆԵՐ
ДОКЛАДЫ АКАДЕМИИ НАУК АРМЯНСКОЙ ССР

LXXIX 1984 3

ПРИКЛАДНАЯ МАТЕМАТИКА

УДК 681.141.2

Э. У. Казарян, С. С. Погосян

Об одном методе планирования вычислительных схем

(Представлено академиком АН Армянской ССР Г. М. Гарибяном 19/V 1983)

При разработке специального математического обеспечения систем программирования большой интерес представляют вопросы организации планирования вычислительных схем на основе формализованной предметной области ([1-5]). Во многих областях науки, в частности химической кинетике, при определении наиболее вероятного механизма сложных цепных реакций возникает необходимость учета в процессе планирования условий, налагаемых на параметры модулей, в качестве которых выступают элементарные химические реакции, определяемые следующим образом:

— входами модулей являются исходные вещества соответствующих химических реакций, а выходами—продукты этих реакций;

— все входы и выходы именованы;

— на выходные параметры модуля налагается предикатное условие (например, свойство радикальности вещества—в спланированной группе реакции указанное вещество должно обязательно быть входом какой-либо реакции данной группы).

Ниже описываются алгоритмы процесса планирования, реализованные в базовой диалоговой системе АНИ-81.

Пусть заданы:

1) конечное множество $\mathbf{X}=\{x_1, x_2, \ldots\}$;

2) конечное множество $F=\{f_1, f_2, \ldots f_i \ldots\}$ операций арности $m_i \times n_i$, m_i, $n_i \geqslant 1$ для представления модулей;

3) с каждой операцией $f \in F$ арности $m \times n$ связаны наборы $\mathrm{In}(f) \subseteq \mathbf{X}$ входных и выходных $\mathrm{Out}(f) \subseteq \mathbf{X}$ аргументов соответственно мощностей m и n.

Пару $(\mathbf{X}, F)$ назовем вычислительной моделью. Пусть $X \subseteq \mathbf{X}$. Определим множество $T(X, F)=\{t, r, \ldots\}$ термов, порожденных множеством $X \subseteq \mathbf{X}$. С каждым $t \in T(X, F)$ определим множества $\mathrm{In}(t)$, $\mathrm{Out}(t)$, список $S(t)$ и глубину терма $d(t)$:

1) если $f \in F$, то $t \in T(X, F) \rightleftarrows \mathrm{In}(f) \subseteq X$; определяем:

$$\mathrm{In}(t)=\mathrm{In}(f),\ \mathrm{Out}(t)=\mathrm{Out}(f),$$

$$S(t)=\langle f \rangle,\ d(t)=1;$$

2) пусть $t \in T(X, F)$ и с t связаны $\mathrm{In}(t)$, $\mathrm{Out}(t)$, $S(t)$, $d(t)$; пусть $f \in F$ и f не входит в список $S(t)$. Тогда

$$t \circ f \in T(X, F) \rightleftarrows \mathrm{In}(f) \subseteq \mathrm{Out}(t) \cup X;$$

определяем:

$$\mathrm{In}(t\circ f)=\mathrm{In}(t)\bigcup(\mathrm{In}(f)\setminus\mathrm{Out}(t)),\ \mathrm{Out}(t\circ f)=\mathrm{Out}(t)\bigcup\mathrm{Out}(f),$$

$$S(t\circ f)=S(t)\dot{\oplus}\langle f\rangle,\ d(t\circ f)=d(t)+1;$$

символ $\dot{\oplus}$ обозначает добавление с конца к списку $S(t)$ операции f.

Если существует терм t такой, что $S(t)=\langle f_{i_1}, f_{i_2}, \dots f_{i_k}\rangle$, то t называется термом, порожденным списком $S(t)$.

Для списка $S(t)=\langle f_{i_1}, f_{i_2}, \dots f_{i_k}, \dots f_{i_p}\rangle$ под операцией $S(t)\setminus\langle f_{i_k}\rangle$ будем предполагать список $\langle f_{i_1}, \dots f_{i_{k-1}}, f_{i_{k+1}}, \dots f_{i_p}\rangle$. Обозначим через $S^*(t)$ множество операций списка $S(t)$. Произвольную пару (X, Y), где $X, Y\subseteq\mathbf{X}$, назовем задачей, если $Y\setminus X\neq\varnothing$.

Пусть термы $t, r\in T(X, F)$. Терм является подтермом r, если $S^*(t)\subseteq S^*(r)$ (обозначается $t\prec r$). Два терма $t, r\in T(X, F)$ называются эквивалентными, если $t\prec r\wedge r\prec t$, т. е. $S^*(t)=S^*(r)$ (обозначается $t\sim r$).

С каждым термом $t\in T(X, F)$ связывается предикат $P_t(x)$, $\forall x\in\mathrm{Out}(t)$, причем: если $t\prec r$, то $P_t(x)=1\to P_r(x)=1$, $\forall x\in\mathrm{Out}(t)$.

Замечание. Из определения вытекает, что:

если $t\sim r$, то $P_t(x)=P_r(x)$, $\forall x\in\mathrm{Out}(t)$;

если $t\prec r$, то $P_r(x)=0\to P_t(x)=0$, $\forall x\in\mathrm{Out}(r)$.

Будем говорить, что задача (X, Y) разрешима в вычислительной модели $(\mathbf{X}, F)$, если существует терм $t\in T(X, F)$ такой, что $\mathrm{Out}(t)\supseteq Y$ и $P_t(x)=1$, $\forall x\in\mathrm{Out}(t)$. При этом терм t называется вычислительной схемой задачи (X, Y).

Обозначим через $\tilde{T}(X, Y, F)$ множество всех вычислительных схем задачи (X, Y). Вычислительная схема t задачи (X, Y) называется тупиковой, если не существует вычислительной схемы r задачи (X, Y) такой, что $S^*(r)\subset S^*(t)$.

Вычислительную схему $t\in\tilde{T}(X, Y, F)$ назовем замыкающей, если $d(t)=\max\limits_{r\in\tilde{T}(X, Y, F)} d(r)$, и соответственно минимальной, если $d(t)=\min\limits_{r\in\tilde{T}(X, Y, F)} d(r)$.

Задача нахождения минимальной вычислительной схемы является алгоритмически неразрешимой.

Задача 1. Планирование (построение) замыкающей вычислительной задачи (X, Y) в вычислительной модели $(\mathbf{X}, F)$.

Задача 2. Планирование тупиковой вычислительной схемы задачи (X, Y) в вычислительной модели $(\mathbf{X}, F)$.

Пусть требуется решить задачу (X, Y).

Теорема. *Для того чтобы вычислительная схема* $t^*\in\tilde{T}(X, Y, F)$ *являлась замыкающей, необходимо и достаточно, чтобы*

$$t\prec t^*,\ \forall t\in\tilde{T}(X, Y, F).$$

Доказательство. Достаточность очевидна.

Необходимость. Пусть t не является подтермом t^* и

$$S(t)=\langle f_{i_1}, \dots f_{i_v}\rangle,\ v\geq 1,\ S(t^*)=\langle f_{j_1}, \dots f_{j_l}\rangle,\ l\geq 1.$$

Рассмотрим список $S'=\langle f_{k_1}, \dots f_{k_p}\rangle$, составленный из тех операций списка $S(t)$, которые не встречаются в списке $S(t^*)$, с сохранением их порядка в списке $S(t)$. В силу предположения $p\geqslant 1$. Рассмотрим список $S''=S(t^*)\oplus S'$. Легко доказать, что список S'' порождает терм t' с $d(t')=d(t^*)+p$ и $t'\in\bar{T}(X, Y, F)$, тогда $d(t')=\nu+p>d(t^*)$, что противоречит определению замыкающей вычислительной схемы. Следовательно $t\prec t^*$.

Следствие. Замыкающая вычислительная схема задачи (X, Y) единственна с точностью до эквивалентности,

Обозначим через $R(t, 0)=\{f\in S^*(t)/\exists x\in \mathrm{Out}(t),\ P_t(x)=0\}$.

Алгоритм построения замыкающей вычислительной схемы задачи (X, Y) представляется в следующей последовательности шагов:

1) полагаем список $\tilde{M}$ пустым;

2) ставится задача (X', Y), где $X'=X$;

3) выбирается множество всех термов $M\subseteq T(X', F)$, для которых глубина равна 1;

4) если $M=\varnothing\wedge\tilde{M}=\varnothing$, то переход на 12);

5) если $M=\varnothing$, то переход на 8);

6) полагаем $\tilde{M}=\tilde{M}\oplus M$ (добавляются справа элементы множества M в произвольном порядке);

7) полагаем $X'=X'\bigcup\limits_{t\in M}\mathrm{Out}(t)$, $F'=F'\setminus M$; переход на 3);

8) порождается терм t_0 списком $\tilde{M}$;

9) если $Y\subseteq\mathrm{Out}(t_0)$, то переход на 12);

10) если $R(t_0, 0)\neq\varnothing$, то задача разрешима и терм t_0 является вычислительной схемой задачи (X, Y); переход на 12);

11) $F=S^*(t_0)\setminus R(t_0, 0)$; переход на 1);

12) задача (X, Y) не разрешима;

13) конец алгоритма.

Пусть в процессе функционирования алгоритма на шаге 8 порождена последовательность термов $t_0^1, t_0^2, \ldots t_0^p$, а на шаге 11 —множества $F_1, F_2, \ldots F_p$, где t_0^i соответствует терму t_0 и F_i множеству F, построенные при i-ом проходе алгоритма $(F_1=F)$. Тогда имеет место следующая лемма: *если $t^*\in\bar{T}(X, Y, F)$ — замыкающая вычислительная схема, то*

$$t^*\prec t_0^i,\ i=\overline{1, p}.$$

Используя предыдущую теорему и лемму, легко доказать, что алгоритм является корректным.

Алгоритм построения тупиковой вычислительной схемы задачи (X, Y) представляется в следующей последовательности шагов:

1) полагаем список $S_0=\varnothing$, $j=1$, $X'=X$, $F'=F$;

2) функционирует алгоритм построения замыкающей вычислительной схемы t^* задачи (X, Y) в вычислительной модели (X, F');

3) если t^* не существует, то переход на 7);

4) полагаем $j=2$, $F'=S^*(t^*)\setminus \mathrm{begin}S(t^*)$, где $\mathrm{begin}S(t^*)$ означает 1-ый по порядку элемент списка $S(t^*)$;

5) если множество $F'\neq\varnothing$, то переход на 2);

6) список $S_0\oplus\mathrm{begin}\,S(t^*)$ порождает тупиковую вычислительную схему задачи (X, Y) в вычислительной модели (X, F); переход на 9);

7) если $j>1$, то $S_0=S_0 \oplus \mathrm{begin}\, S(t^*)$, $X'=X' \bigcup \mathrm{Out}(\mathrm{begin}\, S(t^*))$ и переход на 2);

8) задача (X, Y) не разрешима;

9) конец алгоритма.

Вышеуказанный алгоритм построения замыкающей вычислительной схемы задачи (X, Y) позволяет решать также ряд других задач процесса планирования.

Вычислительный центр
Академии наук Армянской ССР
и Ереванского государственного университета

Է. Հ. ՂԱԶԱՐՅԱՆ, Ս. Ս. ՊՈՂՈՍՅԱՆ

Հաշվարկային սխեմաների կառուցման մի մեթոդի մասին

Աշխատանքում դիտարկված են «Անի—81» երկխոսական գիտական հետազոտությունների ավտոմատիզացման համակարգում իրագործված հաշվարկային սխեմաների բացահայտման հիմնական ալգորիթմները։ Բերված են եզրափակող և փակուղային հաշվարկային սխեմաների կառուցման ալգորիթմները։ Ապացուցված է հետևյալ թեորեմը՝

Որպեսզի $t^*\in T(X, Y, F)$ *հաշվարկային սխեման լինի եզրափակող, անհրաժեշտ է և բավարար, որ տեղի ունենա հետևյալ պայմանը՝*

$$\forall t\in \bar{T}(X, Y, F) \rightarrow t \prec t^*.$$

Աշխատանքում նկարագրված ալգորիթները հիմնված են այդ թեորեմի և նրանից բխող հետևյալ հետևանքի վրա՝

Հետևանք՝ *Հաշվարկային եզրափակող սխեման միակն է համարժեքության ճշտությամբ։*

ЛИТЕРАТУРА — ԳՐԱԿԱՆՈՒԹՅՈՒՆ

[1] *К. А. Абгарян, в сб.:* Мат. вопросы кибернетики и вычислительной техники, вып. XI, Ереван, 1982. [2] *Э. Х. Тыугу,* Вычислительная математика и мат. физика, т. II, № 14 (1971). [3] *Ю. А. Бухштаб, А. И. Горлин* и др., Программирование, № 3, 1981. [4] *В. А. Вальковский,* Программирование, № 6, 1980. [5] *Э. Х. Тыугу,* Программирование, № 4, 1980.