АРХИТЕКТУРА ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ С ПРОГРАММИРУЕМОЙ СТРУКТУРОЙ
(Вычислительные системы)
1980 год Выпуск 82

УДК 681.325.5

РАСПОЗНАВАНИЕ НЕПРОТИВОРЕЧИВОСТИ АЛГОРИТМОВ СТАЦИОНАРНЫХ ПОДСТАНОВОК

С.Н.Сергеев

При построении параллельных микропрограмм на основе алгоритмов параллельных подстановок [1] возникает необходимость выявления ситуаций, когда преобразование,задаваемое микропрограммой,при некоторых исходных данных,неоднозначно.

Алгоритм параллельных подстановок, для которого такая ситуация возможна, называется противоречивым. Определение противоречивости алгоритмов особенно важно при синтезе сети автоматов, интерпретирующей данный алгоритм,поскольку сеть может быть построена только для непротиворечивых алгоритмов [2]. В настоящей работе для подкласса класса алгоритмов стационарных подстановок сформулированы необходимые и достаточные условия непротиворечивости. Также показано, что класс непротиворечивых алгоритмов входит в класс стационарных алгоритмов.

1. Основные понятия и обозначения

Пусть A - конечный алфавит, M - множество имен с мощностью не более чем счетной. Клеткой m, которая находится в состоянии a, называется пара $(a,m) \in A \times M$. В алгоритмах параллельных подстановок основным объектом преобразования является множество (слово) W - конечное множество клеток, в котором нет клеток с одинаковыми именами. Обозначим через $K(A,M)$ совокупность всех клеточных множеств.

При построении алгоритмов параллельных подстановок рассматриваются отображения $S:M \to K(A,M)$. Каждое такое отображение задается набором пар $S = \{(a_1,\varphi_1(m)),\dots,(a_k,\varphi_k(m))\}$,где $a_i \in A$,

$\varphi_i: M \to M$, такие что для $i \neq j$ $(i,j = 1,2,\ldots,k)$ $\varphi_i(m) \neq \varphi_j(m)$ для всех $m \in M$. Область значений отображения S будем называть к о н ф и г у р а ц и е й и обозначать также как S. Таким образом, каждая конфигурация $S = \{(a_1,\varphi_1(m)),\ldots,(a_k,\varphi_k(m))\}$ задает совокупность k-клеточных элементов из $K(A,M)$ вида: $W = \{(a_1, m_1), (a_2,m_2),\ldots,(a_k,m_k)\}$, где $m_1 = \varphi_1(m_1)$, $m_2 = \varphi_2(m_1),\ldots, m_k = \varphi_k(m_1)$. Клеточное множество W будем называть элементом конфигурации S при $m = m_1$ и обозначать как $S(m_1)$. Из двух конфигураций $S_1 = \{(a_1,\varphi_1(m)),\ldots,(a_{n_1},\varphi_{n_1}(m))\}$ и $S_2 = \{(b_1,\psi_1(m)),\ldots,(b_{n_2},\psi_{n_2}(m))\}$ таких, что для $\forall m \in M$, $\forall i,j$ $(i=1,\ldots,n_1$, $j = 1,\ldots,n_2)$, $\varphi_i(m) \neq \psi_j(m)$, можно составить новую конфигурацию $S = \{(a_1, \varphi_1(m)),\ldots,(a_{n_1},\varphi_{n_1}(m)),(b_1,\psi_1(m)),\ldots,(b_{n_2},\psi_{n_2})\}$ называемую п р о и з в е д е н и е м конфигураций S_1 и S_2. Произведение будем обозначать как $S_1 * S_2$. Отметим одно важное свойство произведения конфигураций – ассоциативность: $S = S_1 * S_2 * S_3 = (S_1 * S_2) * S_3 = S_1 * (S_2 * S_3)$, т.е. данную конфигурацию можно представить как произведение конфигураций несколькими способами. Для данной конфигурации $S = \{(a_1,\varphi_1),\ldots,(a_k,\varphi_k)\}$ через $Pr_1(S)$ будем обозначать совокупность состояний $(a_1,\ldots,a_k)$ и называть ее п е р в о й п р о е к ц и е й конфигурации S, совокупность функций $(\varphi_1, \ldots, \varphi_k)$ будем обозначать $Pr_2(S)$ и называть в т о р о й п р о е к ц и е й конфигурации S. При перечислении компонент проекций конфигурации, необходимо придерживаться того порядка, который принят в записи самой конфигурации:

$$\begin{array}{ccccccc}
Pr_1(S) = (& a_1 & , & a_2 & , & a_k &) \\
 & \uparrow & & \uparrow & & \uparrow & \\
S = \{ & (a_1,\varphi_1(m)), & & (a_2,\varphi_2(m)), & \ldots, & (a_k,\varphi_k(m)) & \} \\
 & \downarrow & & \downarrow & & \downarrow & \\
Pr_2(S) = \{ & \varphi_1(m) & & \varphi_2(m) & & \varphi_k(m) & \}
\end{array}$$

Конфигурация всегда однозначно восстанавливается по своим проекциям.

Выражение вида: $S_1 * S_2 \to S_3$, где S_1,S_2,S_3 – конфигурации, называется п о д с т а н о в к о й. Конфигурация $S = S_1 * S_2$ называется левой частью, S_3 – правой частью, S_2 – контекстом. Процесс переработки клеточного множества $W \in K(A,M)$ подстановкой $S_1 * S_2 \to S_3$ заключается в следующем. Если существуют $m_1,m_2,\ldots, m_p \in M$, такие, что $S(m_1), S(m_2),\ldots, S(m_p) \subseteq W$, то подстановка считается применимой к W и результат применения подстановки есть клеточное множество

$$\{W \setminus \bigcup_{i=1}^{p} S_1(m_i)\} \cup \{\bigcup_{i=1}^{p} S_3(m_i)\}.$$

Если таких имен не существует, подстановка неприменима к W. Конечное множество подстановок Φ, записанных в произвольном порядке, называется **системой параллельных подстановок**. Применение системы к $W \in K(A,M)$ задается итерационной процедурой. Пусть W^{i-1} – результат $(i-1)$-й итерации. Если ни одна из подстановок неприменима к W^{i-1}, то W^{i-1} является результатом применения Φ к W. Если же какие-то подстановки применимы (будем обозначать их символами $\Pi_1, \Pi_2, \ldots, \Pi_k$), то результатом i-й итерации применения Φ к W является клеточное множество

$$W^i = \{W^{i-1} \setminus \bigcup_{j=1}^{k} (\bigcup_{i=1}^{p_j} S_1^j(m_i^j))\} \cup \{ \bigcup_{j=1}^{k} (\bigcup_{i=1}^{p_j} S_3^j(m_i^j))\}.$$

Система Φ вместе с такой процедурой применения называется **алгоритмом параллельных подстановок**. При применении алгоритма Φ к некоторому $W \in K(A,M)$ может оказаться, что результат не является клеточным множеством, т.е. содержит клетки с одинаковыми именами, но в разных состояниях.

Например, применяя алгоритм $\Phi: \{(a,x),(a, x+1)\} * (a, x+2) \to \{(b,x),(c,x+1)\}$ к $W = \{(a,1),(a,2),(a,3),(a,4)\}$, получим результат $\Phi(W) = \{(b,1),\underline{(c,2)},\underline{(b,2)},(c,3),(a, 4)\}$, не являющийся клеточным множеством. В таких случаях будем говорить, что алгоритм Φ противоречив относительно W. Для каждой конкретной пары (Φ,W) проверка непротиворечивости (или противоречивости) не представляет труда (путем непосредственного применения алгоритма Φ к W). Однако когда нас интересует непротиворечивость относительно любого множества из $K(A,M)$, необходим критерий, позволяющий только по виду алгоритма определить его непротиворечивость. Не для всякого алгоритма это можно сделать. Однако существует практически важный класс, для которого такой критерий может быть установлен – это класс алгоритмов стационарных подстановок. Подстановка $\Pi: S_1 * S_2 \to S_3$ называется **стационарной**, если $Pr_2(S_1) = Pr_2(S_3)$. Алгоритм Φ, состоящий только из стационарных подстановок, будем называть алгоритмом стационарных подстановок.

Рассмотрим алгоритм стационарных подстановок

$$\Phi : \begin{cases} S_{11} * S_{12} \to S_{13}, \\ S_{21} * S_{22} \to S_{23}, \\ \vdots \\ S_{n1} * S_{n2} \to S_{n3}, \end{cases} \tag{I}$$

где все S_{i1}, S_{i3} – одноклеточные конфигурации вида $\{ (a_i, x) \}$. Алгоритм вида (I) будем называть неймановским алгоритмом.

ТЕОРЕМА I. Неймановский алгоритм непротиворечив тогда и только тогда, когда выполнено одно из условий:

а) $n = 1$;

б) $\forall i, j,\ i \neq j,\ S_{i1} * S_{j1}$ не является конфигурацией;

в) $\forall i, j$ из того, что $S_{i1} * S_{j1}$ – конфигурация, а $S_{i3} * S_{j3}$ не конфигурация, следует, что $S_{i2} * S_{j2}$ не является конфигурацией.

ДОКАЗАТЕЛЬСТВО необходимости проведем от противного. Пусть Φ непротиворечив, а условия "б" и "в" не выполняются[ж)], т.е. существуют i, j такие, что $S_{i1} * S_{j1}$ – конфигурация, $S_{i3} * S_{j3}$ не конфигурация и $S_{i2} * S_{j2}$ – конфигурация.

Возьмем клеточное множество W, содержащее элементы конфигурации $S = (S_{i1} * S_{j1}) * (S_{i2} * S_{j2})$. В качестве W можно, например, взять $S(x)$ при таком x, при котором $S_{i3} * S_{j3}$ не является конфигурацией. Применив W-алгоритм к Φ, получим, что $\Phi(W)$ является клеточным множеством (так как $S_{i3} * S_{j3}$ не конфигурации), что противоречит предположению о непротиворечивости Φ.

Докажем достаточность. Пусть выполнено одно из условий "а"–"в" и нашлось $W = \{(a_1, m_1), \ldots, (a_n, m_n)\}$, такое что $\Phi(W)$ не является клеточным множеством. Следовательно, $\Phi(W)$ содержит хотя бы одну пару клеток (a_i, m_i) и (a_j, m_j) такую, что $m_i = m_j$ $a_i \neq a_j$. Из условия стационарности Φ следует, что на некотором шаге выполнения алгоритма Φ по крайней мере две подстановки (так как любая одна непротиворечива) были применены. Пусть это были Π_i и Π_j.

ж) Доказательство для случая $n = I$ опускаем, ввиду его тривиальности. Неймановский алгоритм, состоящий из одной подстановки, всегда непротиворечив.

Тогда это означает, что $S_{i1}*S_{j1}$ – конфигурация, $S_{i3}*S_{j3}$ не конфигурация, а $S_{i2}*S_{j2}$ – конфигурация. Полученное противоречие доказывает теорему I, которая решает вопрос о непротиворечивости неймановского алгоритма. Покажем, что алгоритм стационарных подстановок при некоторых условиях можно свести к неймановскому с сохранением непротиворечивости.

Пусть дан алгоритм, состоящий из одной подстановки:

$$\Phi: \{(a_0,x),(a_1,\varphi_1(x)),\dots,(a_n,\varphi_n(x))\}*\{(b_1,\psi_1(x)),\dots,(b_m,\psi_m(x))\}\to$$
$$\to \{(c_0,x),(c_1,\varphi_1(x)),\dots,(c_n,\varphi_n(x))\}$$

такой, что все функции $\varphi_i(x)$ имеют обратные.

Сопоставим ему неймановский алгоритм Φ' :

$$\Phi': \begin{cases} S_{11} * S_{12} \to S_{13}, \\ \vdots \\ S_{n1} * S_{n2} \to S_{n3}, \end{cases}$$

получаемый из Φ следующим образом.

Первый шаг. Полагаем $x=y$. Далее, $S_{11}=(a_0,y)$, $S_{13}=(c_0,y)$, $S_{12}=\{(a,\varphi_1(y)),\dots,(a_n,\varphi_n(y))(b_1,\psi_1(y)),\dots,(b_m,\psi_m(y))\}$.

i-й шаг. Полагаем $\varphi_i(x)=y$, во все остальные функции подставляем $x=\varphi_i^{-1}(y)$. Далее, $S_{i1}=(a_i,y)$, $S_{i3}=(c_i,y)$, $S_{i2}=\{(a_0,\varphi_0(\varphi_i^{-1}(y))),(a_1,\varphi_1(\varphi_i^{-1}(y))),\dots,(a_n,\varphi_n(\varphi_i^{-1}(y)))(b_1,\psi_1(\varphi_i^{-1}(y))),\dots,(b_m,\psi_m(\varphi_i^{-1}(y)))\}$ и т.д. до $\varphi_n(x)=y$.

ПРИМЕР.

$$\Phi: \{(a,i),(b,i+1),(c,i+3)\} * \{(c,i-1),(d,i-2)\}\to$$
$$\to\{(c,i),(b,i+1),(a,i+3)\},$$

$$\Phi: \begin{cases} (a,i) * \{(b,i+1),(c,i+3),(c,i-1),(d,i-2)\}\to(c,i), \\ (b,i) * \{(a,i-1),(c,i+2),(c,i-2),(d,i-3)\}\to(b\ i), \\ (c,i) * \{(a,i-3),(b,i-2),(c,i-4),(d,i-5)\}\to(a,i). \end{cases}$$

Для данной подстановки $S_1*S_2\to S_3$ и соответствующего ей неймановского алгоритма Φ': $S_{i1}*S_{i2}\to S_{i3}$ справедлива следующая

ЛЕММА. Для любого $W\in K(A,M)$, если $S_1*S_2(x_0)=W_1\subseteq W$, то и $S_{i1}*S_{i3}(\varphi_i(x_0))=W_1\subseteq W$ $(i=1,\dots,n)$.

ДОКАЗАТЕЛЬСТВО следует из соотношения $\varphi_i^{-1}(\varphi_i(y))=y$, инвариантности конфигурации относительно порядка записи компонент и свойства ассоциативности произведения конфигураций.

В самом деле, полагая $y = \varphi_i(x_0)$ и подставляя в $S_{i1} * S_{i2}(x)$, получим элемент конфигурации $(a_i, \varphi_i(x_0)) * \{(a_0, x_0), (a_1, \varphi_1(x_0)), \ldots, (a_{i-1}, \varphi_{i-1}(x_0)), (a_{i+1}, \varphi_{i+1}(x_0)), \ldots, (a_n, \varphi_n(x_0))(b_1, \phi_1(x_0)), \ldots, (b_m, \phi_m(x_0))\}$, отличающийся от элемента конфигурации $S_1 * S_2(x_0)$ только порядком записи и положением символа "$*$", т.е. имеет место равенство

$$S_1 * S_2(x_0) = S_{i1} * S_{i2}(\varphi_i(x_0)) \subset W, \qquad (2)$$

доказывающее лемму.

Из леммы непосредственно следует одно важное свойство алгоритма Φ'.

СЛЕДСТВИЕ. Для любого $W \in K(A,M)$ справедливо $S_{i1} * S_{i2}(x_0) \subseteq W \longleftrightarrow S_{j1} * S_{j2}(\varphi_j(\varphi_i^{-1}(x_0))) \subseteq W$ для любых $i,j = 1,\ldots,n$, $S_{i1} * S_{i2}$ и $S_{j1} * S_{j2}$ из Φ'.

ДОКАЗАТЕЛЬСТВО. Поскольку равенство (2) справедливо для любых i $(1,\ldots,n)$, то имеем

$$S_{i1} * S_{i2}(\varphi_i(x_0)) = S_{j1} * S_{j2}(\varphi_j(x_0)) \qquad (3)$$

и, полагая $\varphi_i(x_0) = y_0$, получаем

$$S_{i1} * S_{i2}(y_0) = S_{j1} * S_{j2}(\varphi_j(\varphi_i^{-1}(y_0))) .$$

Пусть $\Phi: S_1 * S_2 \to S_3$ и Φ' – неймановский алгоритм, соответствующий Φ. Тогда имеет место

ТЕОРЕМА 2. Справедливо соотношение

$$\bigcup_{i=1}^{n} S_{i3}(\varphi_i(x_0)) = S_3(x_0). \qquad (4)$$

ДОКАЗАТЕЛЬСТВО аналогично доказательству леммы в силу правил применения подстановок.

Будем говорить, что алгоритмы Φ_1 и Φ_2 сильно эквивалентны, если для любого $W \in K(A,M)$ на каждом шаге применения алгоритмов Φ_1 и Φ_2 к W имеет место равенство $\Phi_1(W^i) = \Phi_2(W^i)$, где i – номер шага. Пусть Φ – некоторый алгоритм и Φ' – алгоритм, полученный из Φ заменой каждой подстановки неймановским алгоритмом путем применения описанной выше процедуры. Тогда непосредственным следствием из леммы и теоремы 2 является

ТЕОРЕМА 3. Алгоритмы Φ и Φ' сильно эквивалентны.

Следовательно, Φ непротиворечив тогда и только тогда, когда непротиворечив Φ'.

Рассмотрим теперь вопрос о непротиворечивости любой заданной системы параллельных подстановок. Верна следующая

ТЕОРЕМА 4. Непротиворечивый алгоритм стационарен.

ДОКАЗАТЕЛЬСТВО. Пусть Φ – непротиворечивая система подстановок. Покажем, что она стационарна. Предположим, что это не так, тогда Φ содержит хотя бы одну нестационарную подстановку. Пусть это будет подстановка

$$\{(a_0,\varphi_0),\dots,(a_{n_1},\varphi_{n_1})\}*\{(b_0,\phi_0),\dots,(b_{n_2},\phi_{n_2})\}\to$$
$$\to\{(c_0,f_0),\dots,(c_{n_3},f_{n_3})\}. \qquad (5)$$

Для определенности положим, что $f_0\neq\varphi_i$ $(i=0,\dots,n_1)$. Рассмотрим множество слов $S=\{(a_0,\varphi_0),\dots,(a_{n_1},\varphi_{n_1})(b_0,\phi_0),\dots,(b_{n_2},\phi_{n_2})(c_0^{\prime}\ f_0)\}$, $c_0\neq c_0^{\prime}$. Тот факт, что такое множество существует, следует из того, что $\{(a_0,\varphi_0),\dots,(a_{n_1},\varphi_{n_1})(b_0,\phi_0),\dots,(b_{n_2},\phi_{n_2})\}$ – конфигурация и $f_0\neq\varphi_i$. Взяв любое слово из данного множества $\{(a_0,\varphi_0(x_1)),\dots,(a_{n_1},\varphi_{n_1}(x_1))(b_0,\phi_0(x_1)),\dots,(b_{n_2},\phi_{n_2}(x_1))(c_0^{\prime},f_0(x_1))\}$ и применив к нему подстановку (5), получим слово $\{(c_0,f_0(x_1)),(c_0^{\prime},f_0(x_1)),\dots\}$, не являющееся клеточным множеством, что противоречит предположению.

Таким образом, мы выяснили, что непротиворечивыми относительно всего множества $K(A,M)$ могут быть только стационарные системы подстановок. Однако из этого не следует, что мы не можем при построении алгоритмов использовать нестационарные подстановки. При использовании нестационарных подстановок нужно всегда указывать то подмножество слов $V\subset K(A,M)$, относительно которого они непротиворечивы.

Приведем один пример алгоритма, противоречивого в абсолютном смысле, но непротиворечивого относительно некоторого множества слов.

Пусть $A=(a_0,\dots,a_k,q_0,\dots,q_2,v)$;

$M=\{0,1,2,\dots\}$;

$\varphi_0=x,\ \varphi_1=x+1,\ \varphi_2=x+2,\ \varphi_3=x+3$;

Π_1: $(q_\alpha,x),(a_i,x+1),(a_0,x+2)\to(a_i,x),(q_\beta,x+1),(a_0,x+2)$,

$$\Pi_2: (q_\alpha,x),(a_i,x+1),(a_1,x+2) \to (a_i,x),(q_\beta,x+1),(a_1,x+2),$$

....

$$\Pi_{m+1}:(q_\alpha,x),(a_i,x+1),(a_m,x+2) \to (a_i,x),(q_\beta,x+1),(a_m,x+2),$$

$$\Pi_{m+2}:(q_\alpha,x),(a_i,x+1),(v,x+2) \to (a_i,x),(q_\beta,x+1),(a_0,x+2),(v,x+3).$$

Алгоритм Φ содержит одну нестационарную подстановку Π_{m+2} . Если алгоритм $\Phi\setminus\{\Pi_{m+2}\}$ непротиворечив, то Φ непротиворечив относи - тельно множества слов вида

$$W = \{(u_0,x),(u_1,x+1),\ldots,(u_k,x+k),(v,x+k+1)\}, \quad u_i \neq v .$$

В заключение отметим три важных вопроса, исследование которых поможет решению задачи о распознавании непротиворечивости в общем виде.

Теорема I дает необходимые и достаточные условия непротиво - речивости в предположении, что существуют конструктивные процедуры распознавания конфигураций.Для всех практически важных случаев это предположение справедливо. Но в общем случае для любой заданной совокупности функций необходимо уметь выяснить, можно ли данную совокупность использовать для записи конфигурации.Этот вопрос пока не исследован.

Не исследован также вопрос о существовании аналога теоремы I для стационарных алгоритмов, не сводимых к неймановским.

Третий вопрос относится к поиску конструктивных критериев определения непротиворечивости относительно заданной совокупности клеточных множеств $V \subset K(A,M)$.

Литература

I. БАНДМАН О.Л., ПИСКУНОВ С.В., СЕРГЕЕВ С.Н.Задачи параллельного микропрограммирования - В кн.: Вопросы теории и построения вычислительных систем. (Вычислительные системы, вып. 73.) Новосибирск, 1977, с. 3-24.

2. КОРНЕВ Ю.Н., ПИСКУНОВ С.В., СЕРГЕЕВ С.Н. Алгоритмы обобщенных подстановок и их интерпретация сетями автоматов и однородными машинами. - Изв. АН СССР. Техн. кибернетика, 1971, № 6,с.131-142.

Поступила в ред.-изд.отд.
17 октября 1980 года