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

УДК 681.322.01

РЕДУКЦИЯ СЕТЕЙ ПЕТРИ

П.А. Анишев

Задача анализа сетей Петри на живость и безопасность является экспоненциальной по времени [1-3], поэтому представляется целесообразным сократить исходную сеть, применяя к ней специальные преобразования (правила редукции). Эти преобразования должны не только сохранять живость и безопасность (1-ограниченность) сети, но и иметь полиномиальную сложность. Если сложность преобразований экспоненциальна, то они теряют смысл, так как сама задача анализа экспоненциальна.

В работах [4-6] даны правила редукции, сохраняющие k-ограниченность и живость сети, но для целей создания практических алгоритмов редукции они не все подходят по двум причинам:

1) из сохранения k-ограниченности не следует сохранение безопасности сети (см. пример на рис.4);

2) сложность одного из трех преобразований, введенных в [4-6], является чрезмерно большой, так как для применения этого преобразования нужно найти целочисленное решение системы линейных уравнений и неравенств порядка n, где n - число переходов сети.

Целью данной работы является создание полиномиального алгоритма редукции сетей Петри.

§1. Постановка задачи

Рассматривается класс сильно связных сетей Петри без кратных дуг и с начальным маркированием M_0, которое в каждой позиции содержит не более одной метки. Такие сети будем называть РМ-сетями. Это ограничение (по сравнению с классом обобщенных сетей Петри, см. [7]) существенно упрощает представление данных и расширяет мощностные ограничения при программировании алгоритма редукции и в то же время не ограничивает общности рассмотрения, так как любую обобщенную сеть можно преобразовать в PN-сеть [7]. Живые и безопасные PN-сети назовем правильными. Графически PN-сеть представляется ориентированным двудольным графом: вершины сети - это позиции и переходы, позиции обозначаются кружками, переходы - черточками. Через $A^{\bullet}$ (соответственно $^{\bullet}A$) обозначается множество выходных (входных) вершин множества A. Определение понятий, относящихся к сетям Петри, можно найти в [7].

Задача редукции ставится следующим образом: определить множество преобразований $R = \{R_1, R_2, \ldots, R_n\}$ на классе PN-сетей так, чтобы выполнялись следующие условия:

1. Применение любого $R_i: N \to N'$ сохраняет свойство правильности, т.е. сеть N' правильная тогда и только тогда, когда N - правильная.

2. Для любой PN-сети N существует конечная последовательность правил из R, такая, что к результирующей сети N' никакое R_i неприменимо.

3. Сложность любого R_i не превосходит полинома второй степени.

§2. Правила редукции

Определим четыре типа правил редукции: удаление избыточного перехода, подстановка позиции, удаление избыточной позиции, подстановка перехода.

Рассмотрим их подробнее.

2.1. Удаление избыточного перехода - правила R11 и R12. Переход t_1 называется избыточным, если его удаление не нарушает свойства правильности (или неправильности) сети. Сформулируем (в терминах структурных свойств сети) два правила, при которых это возможно.

П р а в и л о R11 . Если множества входных позиций двух (или более) переходов совпадают, $^{\bullet}t_1 = {}^{\bullet}t_2$, и множества выходных позиций совпадают, $t_1^{\bullet} = t_2^{\bullet}$ (см. рис. 1,а), то один из переходов может быть удален.

П р а в и л о R12 . Если множество входных и выходных позиций перехода t_1 одно и то же, $^{\bullet}t_1 = t_1^{\bullet}$, и существует переход t_2 такой, что $t_1^{\bullet} \subseteq t_2^{\bullet}$ (см. рис.1,б), то переход t_1 может быть удален.

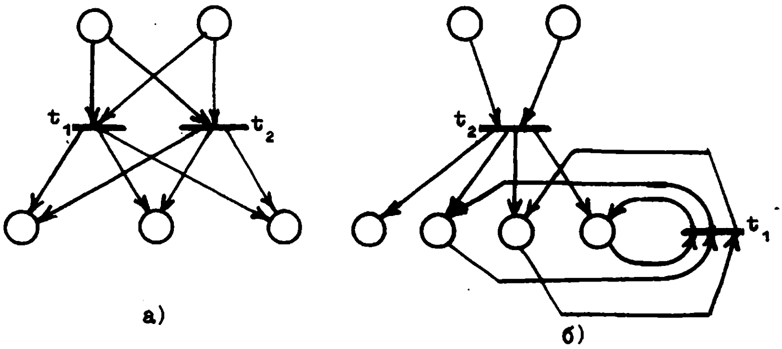


Рис. I

2.2. <u>Подстановка позиции</u> - правило R2 .

Правило подстановки для PN-сетей формулируется так же, как в [5]. Позиция e называется подстановочной тогда и только тогда, когда (см. рис.2,а) существуют два множества переходов $H = \{1,2\}$ и $F = \{3,4,5\}$, удовлетворяющие условиям:

$\forall f \in F$:

I) e является единственным входом в f ;

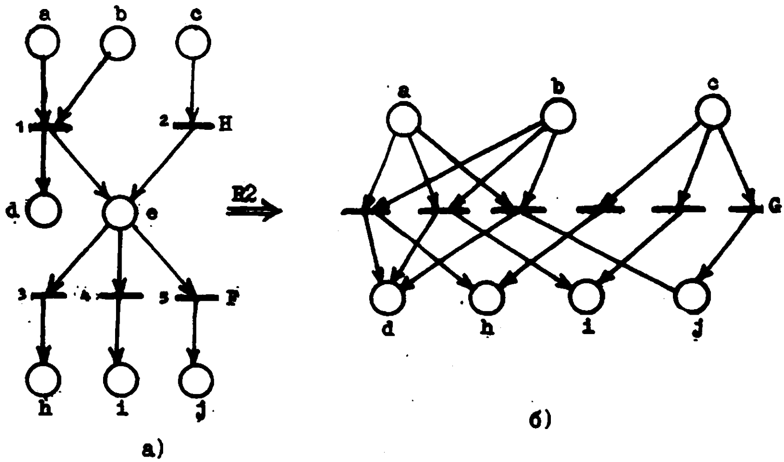


Рис. 2

2) e не является выходом из f;

3) f имеет по крайней мере один выход;

$\forall h \in H$:

e не является входом в h;

$\forall t \notin H \cap F$:

переход t не связан с e.

Подстановка состоит в замене позиции e на h,i,j для каждого из переходов множества H (см. рис. 2,б).

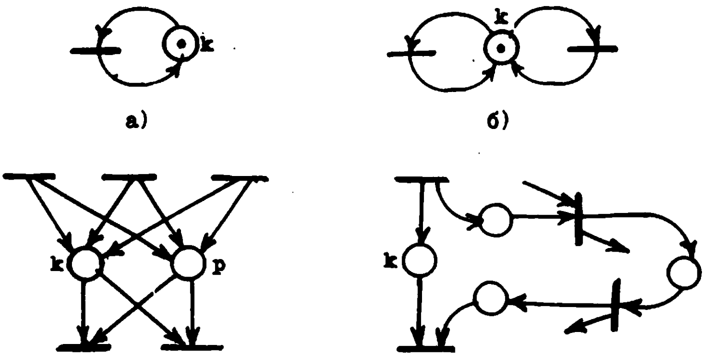
Подстановка позиции e состоит в следующем (см. рис. 2,б):

1. Из сети удаляется позиция e и множества переходов H и F.

2. Добавляется множество переходов G; каждый переход из H заменяется на столько переходов, какова мощность F, т.е. $|G| = |H| \cdot |F|$, при этом для каждого перехода из G выполняется следующее условие:

$$\forall g \in G\ \exists^1 h \in H\ \exists^1 f \in F[{}^\bullet g = {}^\bullet h \ \&\ g^\bullet = (h^\bullet \setminus \{e\}) \cup f^\bullet].$$

2.3. <u>Удаление избыточной позиции</u> - правила R31 - R34. Позиция k называется избыточной, если ее удаление не нарушает свойства правильности (или неправильности) оставшейся сети. Выделим четыре случая: в первых двух маркирование избыточной позиции равно единице при срабатывании любого перехода сети; в третьем и четвертом - число меток в избыточной позиции равно сумме маркирований некоторого фиксированного подмножества позиций при любом маркировании, достижимом из начального.



а) б) в) г)

Рис. 3

П р а в и л о R31 . Если маркированная позиция k имеет один входной и один выходной переходы и эти переходы совпадают, т.е. $k^\bullet = {}^\bullet k$, $M(k) = 1$, $|k^\bullet| = 1$ (см. рис.3,а), то позиция k может быть удалена.

П р а в и л о R32 . Если множества входных и выходных переходов маркированной позиции k совпадают, $k^\bullet = {}^\bullet k$, $M(k) = 1$ (см. рис. 3,б), то позиция k может быть удалена.

П р а в и л о R33 . Если существует позиция p такая, что $k^\bullet = p^\bullet$, ${}^\bullet k = {}^\bullet p$, $M(k)=M(p)$(см. рис. 3,в), то позиция k может быть удалена.

П р а в и л о R34 . Если позиция k имеет один входной и один выходной переходы,причем от входного к выходному переходу существует путь, проходящий через множество позиций A , каждая из которых имеет один вход и один выход, и выполняется соотношение $M(k) = \sum_{q \in A} M(q)$ (см.рис.3,г), то позиция k может быть удалена.

Введение списка правил R31-R34 (вместо одного правила удаления избыточной позиции в [4-6]) обусловлено тем,что (как было замечено во введении) из сохранения k-ограниченности сети не следует сохранения 1-ограниченности. Действительно, рассмотрим пример на рис.4. Позиция e является избыточной в смысле Barthelot[4-6], но она не является безопасной, хотя и является ограниченной (при любых достижимых маркированиях число меток в позиции e не превосходит двух). Таким образом,исходная сеть, изображенная на рис. 4,а, не является правильной, но после удаления позиции e мы получаем правильную сеть (рис.4,б). Правило R31 является частным случаем R32. Введение R31 обусловлено потребностью получения эффективного алгоритма редукции.

Список правил R31-R34 не является исчерпывающим, и можно привести другие правила,но, по нашему мнению, введенные правила охватывают довольно большой процент практически важных случаев редукции, в то же время сложность их полиномиальна (см. п. 3.4).

2.4. Подстановка перехода - правило R4.

По аналогии с правилом подстановки позиции, введем правило подстановки перехода. Переход t_1 (см. рис.5) называется подстановочным, если существуют множество позиций P и позиция r и выполняются следующие условия:

1) позиция r имеет один вход и один выход $r^\bullet = \{t_2\}$, ${}^\bullet r=\{t_1\}$;

2) позиция r не маркирована.

Подстановка перехода состоит в удалении t_1 и r и в добавлении дуг, соединяющих каждую из позиций $q \in P$ с переходом t_2.

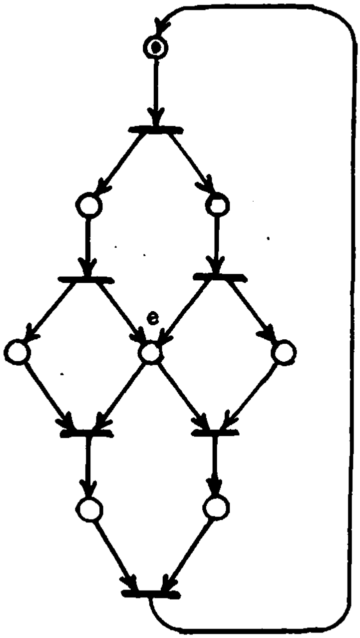


Рис. 4

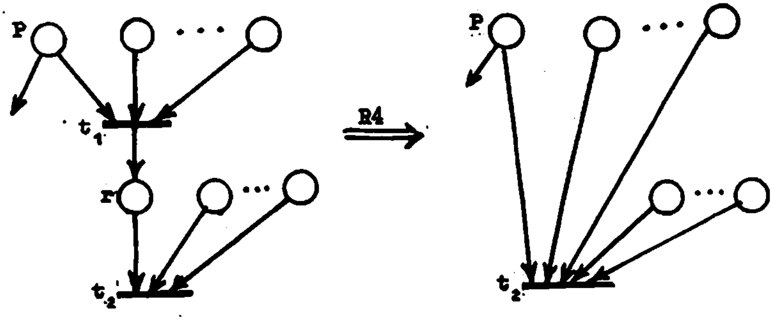


Рис. 5

Доказательство того, что введенные правила R11-R4 сохраняют правильность (или неправильность) сети, можно получить, используя принцип "переноса" свойств живости и безопасности, который заключается в следующем. Если удаляется (или вводится) позиция (переход), то в сети найдется другая позиция (переход), имеющая те же свойства, что и удаленная, при всех маркированиях, достижимых из начального. Например, для перехода t_1 (см.рис.I) таким переходом является t_2 ; для позиции k (рис.3,в) - позиция p ; для позиции k (рис. 3,г) - множество позиций A.

§3. Анализ правил редукции

В данном параграфе обсуждаются свойства множества $R = \{R_1, \ldots, R_n\}$ правил редукции. Через R^* обозначим множество конечных последовательностей элементов из R , через $|N|$ - число позиций сети N , а через $|R_i|$ - временнýю сложность правила R_i .

3.I. Последовательность применения правил из R конечна, если для любой PN-сети N_0 существует $\alpha \in R^*$ такая, что к сети N_m , полученной на последнем шаге, никакое $R_k \in R$ неприменимо. Конечность следует из того, что каждое из правил уменьшает число позиций сети:

$$|N_0| > |N_1| > \ldots > |N_m|.$$

3.2. Правило $R_k \in R$ независимо от множества правил $R \setminus R_k$, если найдется сеть N_0 такая, что результат редукции N_0 без использования правила R_k будет иметь бóльшее число позиций, чем при редукции с использованием всех правил из R , т.е.

$$\exists N_0 \exists \alpha \in R^* \forall \beta \in (R \setminus R_k)^* \; [(N_0 \overset{\alpha}{\to} N' \;\&\; N_0 \overset{\beta}{\to} N'') \to |N'| < |N''|].$$

Для того чтобы доказать независимость множества правил, введенных в §2, достаточно построить для каждого правила пример соответствующей сети N_0 , такой, что к ней никакое правило из R, кроме R_k, неприменимо. Примеры для правил R11 - R4 приведены на рис. 6.

3.3. Однозначность результата при применении правил редукции из множества R на некотором классе $\mathcal{K}$ заключается в том, что сети N' и N" (получающиеся в результате применения двух различных последовательностей из R) оказываются эквивалентными, т.е.

$$\forall N \in \mathcal{K} \forall \alpha, \beta \in R^* [(N \overset{\alpha}{\to} N' \;\&\; N \overset{\beta}{\to} N'') \to N' \equiv N''].$$

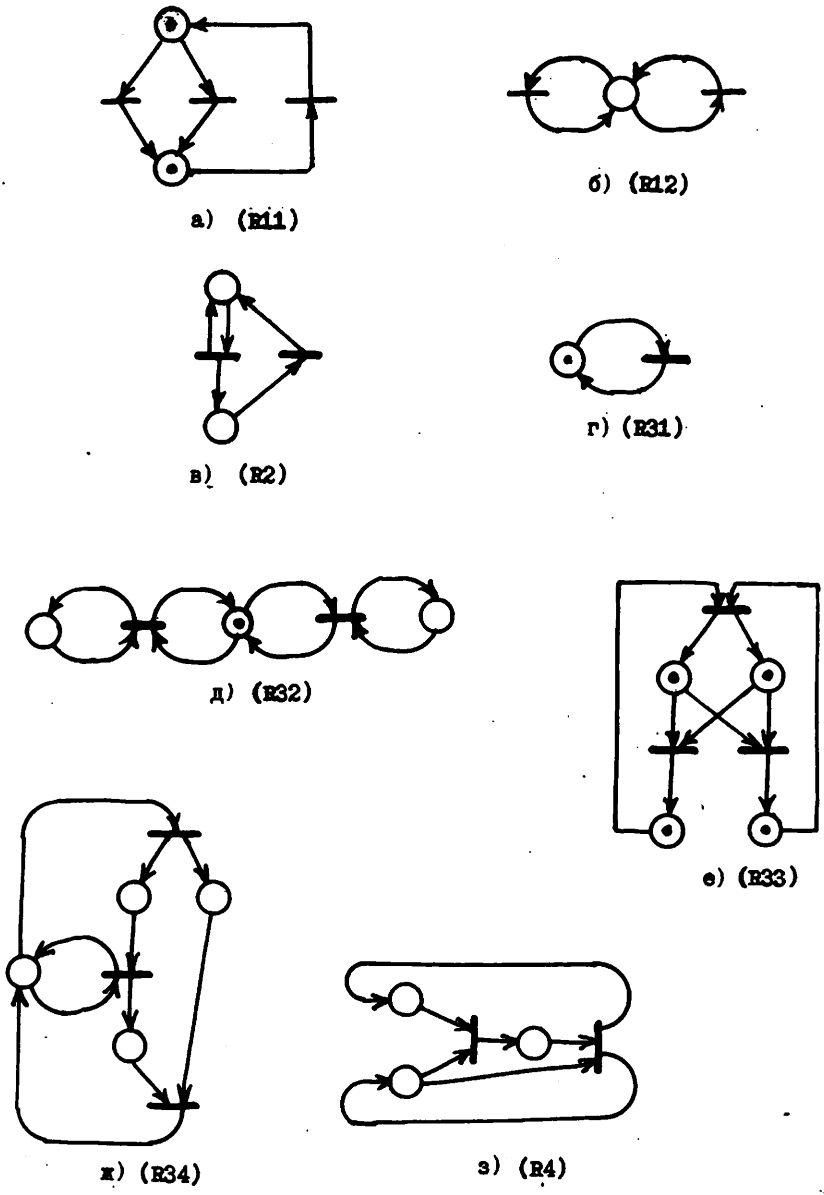


Рис. 6

Под эквивалентностью $\cong$ понимается изоморфизм графов, представляющих сети, причем переходы отображаются в переходы, позиции – в позиции с сохранением маркирования.

Если множество R на классе $\mathscr{H}$ действует однозначно, то знание этого факта избавляет нас от бесплодных поисков наименьшего по числу позиций результата редукции. К сожалению, для многих классов сетей этот факт не имеет места. В [6] показано, что определенные там правила R1, R2, R3, применяемые к обобщенным сетям, не обладают свойством однозначности, хотя для сетей, обладающих свойством Черча-Россера, эти правила действуют однозначно [4].

Для PN-сетей вопрос об однозначности остается открытым, но для маркированных и автоматных графов (так как они редуцируются в сеть с одной позицией или одним переходом, см. [6]) он решен положительно. В §4 будет показано, что PN-сети, получающиеся из хорошо сформированных параллельных граф-схем, редуцируются в сеть с одним переходом без позиций (тривиальная сеть), а, значит, на таких сетях множество правил действует однозначно.

Свойство однозначности позволяет нам не заботиться о порядке применения правил редукции, хотя соображения вычислительной сложности приводят к тому, что сначала желательно использовать менее сложные правила.

3.4. Сложность правила редукции мы будем оценивать числом операций, необходимых для поиска соответствующего фрагмента в сети с n вершинами, пренебрегая сложностью операций замены (или удаления) этого фрагмента. Обозначим через p число позиций сети; t – число переходов; $n = p+t$ – общее число вершин сети; od и id – средние полустепени выхода и входа, а $d = od + id$ – среднюю степень вершин сети; c – некоторая постоянная.

С л о ж н о с т ь R12, R31, R32 и R4 определяется тривиально: $|R12| = ctd$; $|R31| = cp$; $|R32| = cpd$; $|R4| = cp$.

С л о ж н о с т ь R11 и R33. Поиск фрагмента для R11 можно осуществить следующим образом: находим всех соседей очередного перехода ($\sim id$ операций); выделяем для этого перехода входные позиции и находим для них другой общий выходной переход ($\sim id^2$ операций); для найденного перехода проверяем совпадение его выходных позиций с выходными позициями первого перехода ($\sim od^2$ операций). Проведя поиск по всем t переходам, окончательно получим

$$|R11| = ct(id + id^2 + od^2) \leq ct(id + od)^2 = ctd^2 .$$

Рассуждая аналогично (заменяя переходы на позиции, а позиции на переходы), для R33 получим

$$|R33| = cpd^2 .$$

С л о ж н о с т ь R34 . Удаление избыточной позиции должно сохранить сильную связность сети. Для определения сильной связности существует алгоритм с оценкой $O(MAX(n,e))$, где n – число вершин, $e = \frac{nd}{2}$ – число ребер графа [8]. Значит, поиск пути между входными и выходными переходами избыточной позиции с применением этого алгоритма может быть осуществлен за время $cnd/2$. Если в сети имеется p' позиций, имеющих один вход и один выход, то, проверяя поочередно все такие позиции на возможность удаления, получим для R34 следующую оценку:

$$|R34| = cnd\ p'/2 < cn^2 d .$$

С л о ж н о с т ь R2. Для того чтобы найти подставляемую позицию, необходимо проверить, что она является единственным входом в каждый из своих выходных переходов и не является выходом ни для одного из них ($\sim od$ операций). Затем нужно убедиться в том, что эта позиция не является входом ни в один из своих входных переходов ($\sim id$ операций). Таким образом, для R2 получаем

$$|R2| = cp(id + od) = cpd .$$

Эффективность применения правила R_i понимается как число вершин, на которое сокращается сеть при применении R_i . Подстановка позиции уменьшает число позиций, но может увеличить число переходов, поэтому суммарное число вершин может не уменьшиться, а даже возрасти. Если начальное число вершин в окрестности k-й вершины было $od_k + id_k + 1$, то после подстановки позиции k получим $od_k \cdot id_k$ вершин в ее окрестности, таким образом, сеть сокращается на величину Δ_k ,

$$\Delta_k = od_k + id_k + 1 - od_k \cdot id_k ,$$

которая при $od_k + id_k \leq 5$ остается положительной.

Порядок применения преобразований определяется по критерию сложность/эффективность, и то преобразование, для которого это отношение меньше, выполняется первым. Результаты проведенного анализа сведены в таблицу.

Таблица

Правило	Характеристики		
	Сложность	Эффективность	Порядок применения
R11	ctd^2	I	4
R12	ctd	I	3
R2	cpd	I	3
R31	cp	I	2
R32	cpd	I	3
R33	cpd^2	I	4
R34	cn^2d	I	5
R4	cp	2	I

В заключение этого параграфа заметим, что список правил можно пополнять, при этом необязательно вводить независимые правила. При решении этого вопроса нужно учитывать такие характеристики, как сложность, эффективность, однозначность, общее время работы алгоритма редукции, простота программирования и т.п.

§4. Классы сетей Петри

В данном параграфе вводятся подклассы PN-сетей и решаются вопросы вложения этих классов. Рассмотрим следующие классы сетей Петри:

PN - сильно связные сети без петель и кратных дуг, начальное маркирование - не более одной метки в каждой позиции;

SM - PN -сети, каждый переход которых имеет один вход и один выход;

MG - PN -сети, каждая позиция которых имеет один вход и один выход;

WF - PN -сети без маркирования, т.е. с нулевым маркированием, для которых существует живое и безопасное маркирование;

LS - WF -сети с живым и безопасным маркированием;

CP - сети, полученные из корректных параллельных граф-схем алгоритмов по правилам из [9];

WP - сети, полученные из хорошо сформированных параллельных граф-схем алгоритмов по правилам из [9];

R^1 - сети, редуцируемые в сеть с одной немаркированной позицией;

R^0 - сети, редуцируемые в сеть с одним переходом, без позиций (тривиальная сеть).

Диаграммы вложения классов сетей приведены на рис. 7; на рис. 7,а - классы маркированных сетей; а на рис. 7,б - классы немаркированных (с нулевым маркированием) сетей. Все подмножества, указанные на диаграмме, - собственные. $A \to B$ обозначает, что $A \subset B$.

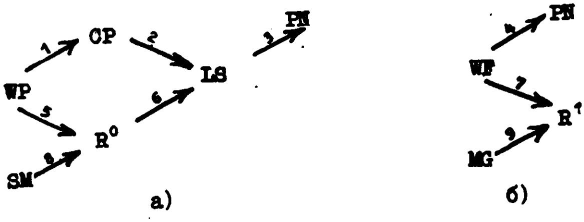


Рис. 7

Вложение I доказано в [9], где приведен пример корректной граф-схемы, не являющейся хорошо сформированной. Вложение 2 составляет суть теоремы, доказанной в работе [IO]. Вложения 3 и 4 очевидны по определению. Вложения 8 и 9 доказаны в [6], причем MG-сети рассматриваются без маркирования и для их редукции используются преобразования R31 и R34. Истинность вложения 6 доказывается путем построения примера правильной сети не редуцируемой в тривиальную. Проведем доказательство.

На рис.8 приведен пример правильной нередуцируемой сети. Действительно, избыточных переходов нет, так как удаление любого перехода нарушает сильную связность (R11 и R12 неприменимы); никакая позиция не может быть подстановочной, так как не является единственным входом в переход (R2 неприменимо); избыточных позиций нет потому, что удаление любой позиции (из соображений симметрии достаточно рассмотреть только две: маркированную и немаркированную) нарушает необходимое и достаточное условие правильности сети, которая является маркированным графом. (Маркированный граф правильный, если и только если каждая дуга принадлежит циклу с одной меткой [11].) При удалении позиции a дуга (e , 2) принадлежит только циклу (e , f , g , h , e), имеющему д в е метки h и f . При удалении позиции e дуга (a,1) принадлежит только циклу (a,b,c,d,a), имеющему тоже д в е метки в позициях a и c (R31-R34 неприменимы). Правило подстановки перехода неприменимо, так как

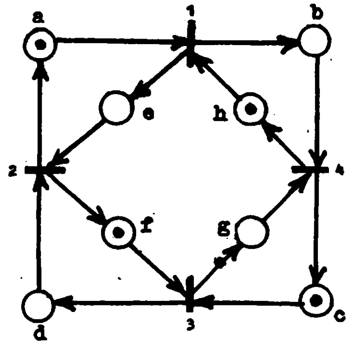


Рис. 8

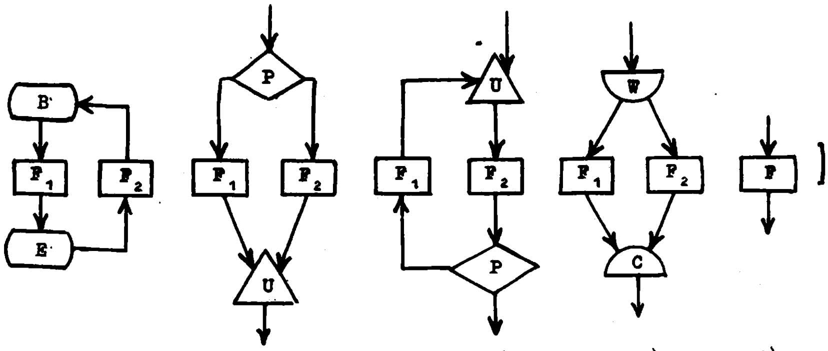


Рис. 9

каждый переход имеет более одного выхода. Окончательно - сеть на рис.8 нередуцируема в смысле правил R11-R4 , но эта сеть является живой и безопасной. Вложение 6 доказано.

Проведем доказательство вложения 5.

Для доказательства вложения 5 (проводимого индукцией по глубине вложения WP-схем) достаточно рассмотреть фрагменты сети Петри, полученные из WP-подсхем следующих пяти типов (см.рис.9,а-д). Легко видеть, что каждый из фрагментов б), в), г) сводится к фрагменту д). Для этого достаточно использовать преобразования R13, R2 , R33. В конце концов мы приходим к сети Петри, соответствующей WP-схеме на рис. 9,а, которая (применением R2 и R31) редуцируется в тривиальную сеть.

Вопрос о вложении 7 остается открытым.

Литература

1. LIPTON R. The reachability problem and the boundness problem for Petri nets are exponential-spase hard.- TR-62, Conn.,Jan., 1976.

2. JONES N.D., LANDWEBER L.H., LIEN Y.E. Complexity of some problems in Petri-nets.- Theoretical computer science,1977, N 4 , p.277-299.

3. RACKOFF C. The covering and boundedness problems for vector addition systems. - Theoretical computer science, 1978, N 6, p.223-231.

4. BERTHELOT G., ROUCAIROL G. Reduction of Petri-nets.- Lecture Notes in Computer Sciences, 1976, N 45, p.202-209.

5. BERTHELOT G. Preuve de non blocage de programmes parallèles par reduction de réseaux de Petri. 1-st European Conf. on Parallel and Distributed Processing.-Toulouse, France, February 14, 16,1979, p.251-259.

6. BERTHELOT G. Vérification de reseaux de Petri. Thèse de 3ème cycle Université Pierre et Marie Curie, Paris, Janvier,1978. -VIII, 212, 8 p.

7. HACK M. Decision problems for Petri nets and vector addition systems.- Computation structures Group Memo 95, Project MAC, M.I.T., Cambridge, Mass., March 1974.-79 p.

8. АХО А., ХОПКОРФТ Дж., УЛЬМАН Дж. Построение и анализ вычислительных алгоритмов. -М.: Мир, 1979. -536 с.

9. АНИШЕВ П.А. О детерминированности параллельных граф-схем. -В кн.: Вопросы теории и построения вычислительных систем (Вычислительные системы, вып. 73.) Новосибирск, 1978, с.40-52.

10. АНИШЕВ П.А. Анализ параллельных граф-схем алгоритмов на корректность с помощью сетей Петри. -Программирование, 1980, №6, с. 82-89.

11. Marked Directed Graphs, [illegible]r F., Holt A.W., Even S., Pnuelli A. - J.of Computer and System Sciences, 1971,v.5, p.511-523.

Поступила в ред.-изд.отд.
29 сентября 1980 года