ПРИКЛАДНАЯ ДИСКРЕТНАЯ МАТЕМАТИКА

2008 Прикладная теория автоматов № 1(1)

ПРИКЛАДНАЯ ТЕОРИЯ АВТОМАТОВ

DOI 10.17223/20710410/1/19

УДК 519.7

К ОПИСАНИЮ ПРОГРЕССИВНЫХ РЕШЕНИЙ ПАРАЛЛЕЛЬНОГО АВТОМАТНОГО УРАВНЕНИЯ[1]

В.Г. Бушков

Томский государственный университет

E-mail: bushkov@sibmail.com

Работа посвящена нахождению наибольшего прогрессивного решения параллельного автоматного уравнения $C \lozenge X \cong S$, где C и S – конечные полностью определенные автоматы, на основе удаления непрогрессивных последовательностей из наибольшего решения уравнения. Именно прогрессивные решения, в композиции которых с автоматом C отсутствуют заведомо тупиковые ситуации, интересны с практической точки зрения.

Ключевые слова: *конечный автомат*, *параллельная композиция*, *автоматное уравнение*, *прогрессивное решение.*

Многие задачи синтеза и анализа систем логического управления сводятся к решению автоматного уравнения $C \lozenge X \cong S$, где C и S – конечные автоматы, $\lozenge$ – операция параллельной композиции, которая соответствует поочередной работе автоматов-компонент [1]. Известно, что разрешимое уравнение имеет наибольшее решение, которое можно рассматривать как резервуар для выбора наилучшего в некотором смысле решения. Однако не каждое решение интересно с практической точки зрения. Особый интерес представляют так называемые прогрессивные решения, т.е. решения, в композиции которых с автоматом C отсутствуют тупики и осцилляции. В работе [2] наибольшее прогрессивное решение строится для случая, когда компоненты уравнения являются полуавтоматами, и предлагается алгоритм нахождения такого решения на основе расщепления состояний наибольшего решения. В данной работе мы предлагаем другой подход к построению наибольшего прогрессивного решения, который основан на удалении из наибольшего решения так называемых не прогрессивных входо-выходных последовательностей. Поскольку в общем случае число последовательностей в наибольшем решении бесконечно, то нетривиальным является вопрос о сходимости предлагаемого алгоритма, и мы показываем, что предложенный алгоритм сходится.

1. Основные определения

Алфавитом A называется непустое конечное множество символов. Обозначим через A^* множество всех конечных слов над алфавитом A. Подмножество $L \subseteq A^*$ называется *языком*. Рассмотрим непустое подмножество A_1 множества A и отображение h: $A \to A_1 \cup \{\varepsilon\}$, где ε – пустое слово. Мы полагаем, что $h(\alpha) = \alpha$ для всех $\alpha \in A_1$ и $h(\alpha) = \varepsilon$ для всех $\alpha \in A \backslash A_1$, причём h расширяется на слова по правилу: $h(\alpha\beta) = h(\alpha)h(\beta)$ и $h(\varepsilon) = \varepsilon$. Тогда язык $L_{\Downarrow A_1} = \{h(\beta) : \beta \in L\}$ называется *ограничением* языка L на алфавит A_1. Пусть теперь язык L определен над алфавитом $A_2 = A \backslash A_1$. Рассмотрим отображение ψ: $A_2 \to 2^{A^*}$, такое, что $\psi(\alpha) = \{\gamma\alpha\beta$: $\gamma, \beta \in A_1{}^*\}$. Тогда язык $L_{\Uparrow A} = \{\psi(\beta)$: $\beta \in L\}$ есть *распространение* языка L на алфавит A. По определению, распространение пустого языка есть пустой язык. Пусть язык L_1 определен над алфавитом A_1, язык L_2 определен над алфавитом A_2, $A = A_1 \cup A_2$ и S – непустое подмножество множества A. Параллельной композицией $L_1 \lozenge_S L_2$ называется язык $(L_{1\Uparrow A} \cap L_{2\Uparrow A})_{\Downarrow S}$. Для языка L над алфавитом A через $Init(L) = \{\alpha \in A^* \mid \exists \beta \in A^*, \alpha\beta \in L\}$ обозначим язык, состоящий из всех префиксов всех слов языка L.

Автоматом называется пятёрка $\mathcal{S} = (S, I, O, T_S, s_0)$, где S – конечное множество состояний, I – конечный входной алфавит, O – конечный выходной алфавит, $T \subseteq I \times S \times S \times O$ – отношение переходов, s_0 – начальное состояние. Автомат называется *полностью определенным*, если $\forall (i,s) \in I \times S\ \ \exists (s',o) \in S \times O\ \ ((i,s,s',o) \in T)$; иначе автомат называется *частичным*. Рассмотрим автоматы $\mathcal{S} = (S, I, O, T_S, s_0)$ и $\mathcal{Q} = (Q, X, Y, T_Q, q_0)$. Если

[1] Работа частично поддержана грантом РФФИ 06-08-89500.

множества $I \cap X$ и $O \cap Y$ не являются пустыми, то пересечением автоматов $S \cap Q$ называется наибольший связный подавтомат автомата $(S \times Q, I \cap X, O \cap Y, H, s_0q_0)$, в котором отношение H определено следующим образом: $(i, sq, s'q', o) \in H \Leftrightarrow (i, s, s', o) \in T_S \;\&\; (i, q, q', o) \in T_Q$.

Полуавтоматом называется пятёрка $M = (M, I, T_M, m_0, Q)$, где M – конечное множество состояний, I – конечный алфавит, $T \subseteq I \times S \times S$ – отношение переходов, m_0 – начальное состояние, $Q \subseteq M$ – подмножество финальных состояний. Языком полуавтомата $L \subseteq I^*$ называется множество всех последовательностей, которые переводят полуавтомат из начального состояния в одно из финальных состояний. Любой автомат $S = (S, I, O, T_S, s_0)$ можно представить полуавтоматом M_S, «растягивая» каждый переход в автомате на два перехода, один из которых помечен входным, а другой – выходным символом. Формально, $M_S = (S \cup S \times I, I \cup O, \Delta, s_0, S)$, где $(i,s,(s,i)) \in \Delta \wedge (o,(s,i),s') \in \Delta \Leftrightarrow (i,s,s',o) \in T_S$. Финальными объявляются состояния из множества S. *Языком* автомата S называется язык соответствующего полуавтомата M, т.е. языком автомата S называется множество входо-выходных последовательностей, допустимых в начальном состоянии автомата. Далее язык полуавтомата M обозначается M. Полуавтомат $Init(M)$ с языком $Init(M)$ получается из полуавтомата M удалением всех состояний, в которых генерируется пустой язык, и объявлением всех оставшихся состояний финальными. Полностью определенные автоматы A и B называются *эквивалентными* ($A \cong B$), если их языки совпадают, т.е. $A \cong B \Leftrightarrow L(A) = L(B)$. Автомат B называется *редукцией* автомата A ($B \le A$), если язык автомата B содержится в языке автомата A, т.е. $B \le A \Leftrightarrow L(B) \subseteq L(A)$. Если из контекста понятно, является рассматриваемый объект автоматом или полуавтоматом, то для простоты изложения автомат и соответствующий ему полуавтомат обозначаются одним символом.

Пусть I_1, I_2, V, U, O_1, O_2 – попарно непересекающиеся конечные множества, некоторые из которых могут быть пустыми. *Параллельной композицией* автоматов A с входным алфавитом $I_1 \cup V$ и выходным алфавитом $O_1 \cup U$ и B с входным алфавитом $I_2 \cup U$ и выходным алфавитом $O_2 \cup V$ называется автомат $A \lozenge_{Ext} B$, далее просто $A \lozenge B$, язык которого есть $(L(A) \lozenge_{Ext} L(B)) \cap (IO)^*$, далее просто $(L(A) \lozenge L(B)) \cap (IO)^*$, где $I = I_1 \cup I_2$, $O = O_1 \cup O_2$, $Ext = I \cup O$[1]. *Автоматным уравнением* называется выражение вида $C \lozenge X \cong S$, где $C = (C, I_1 \cup V, O_1 \cup U, T_C, c_0)$, $S = (S, I_1 \cup I_2, O_1 \cup O_2, T_S, s_0)$ – конечные автоматы, которые соответственно называются *контекстом* и *спецификацией* [2], $X = (X, I_2 \cup U, O_2 \cup V, T_X, x_0)$ – неизвестный автомат, $\lozenge$ – операция параллельной композиции, и $\cong$ – отношение эквивалентности. Как обычно, полностью определенный автомат B называется *решением* уравнения $C \lozenge X \cong S$, если $C \lozenge B \cong S$. Решение M уравнения называется *наибольшим*, если каждое решение уравнения есть редукция автомата M. Известно, что если уравнение $C \lozenge X \cong S$ разрешимо, то уравнение имеет наибольшее решение, которое в общем случае является недетерминированным частичным автоматом с языком $\overline{(L(C)\langle\rangle_{Ext}\overline{(L(S)}\cap(IO)^*)} \cap ((I_2 \cup U)(O_2 \cup V))^*$. В данной работе в качестве коэффициентов уравнения, автоматов C и S, и его решения мы рассматриваем только полностью определенные автоматы, поскольку любая физическая реализация является полностью определенной. Соответственно из наибольшего решения мы итеративно удаляем состояния, в которых не определен хотя бы один переход, и получаем наибольшее полностью определенное решение уравнения [1].

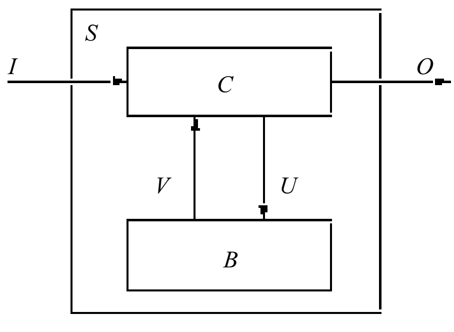

Рис. 1

В качестве примера рассмотрим композицию автоматов на рис. 1. Если поведение композиции описано автоматом $S = (\{s\}, \{i\}, \{o\}, T_S, s)$, где $T_S = \{(i, s, s, o)\}$, и фрагмент композиции, поведение которого описано автоматом $C = (\{c\}, \{i, v\}, \{o, u\}, T_C, c)$, где $T_C = \{(i, c, c, u), (v, c, c, o)\}$, уже синтезирован, то неизвестный фрагмент композиции можно синтезировать как автомат $B = (\{b\}, \{u\}, \{v\}, T_B, b)$, где $T_B = \{(u, s, s, v)\}$, который есть решение автоматного уравнения $C \lozenge X \cong S$.

---

[1] В выходной алфавит O композиции можно добавить символы из внутренних алфавитов U и V, но в данной работе без ограничения общности мы полагаем, что $O = O_1 \cup O_2$.

Решение Sol автоматного уравнения $C \Diamond X \cong S$ называется *прогрессивным*, если Sol полностью определенный автомат, и для любого слова α из языка $Init(L(C))_{\Uparrow(I_2 \cup O_2)} \cap Init(L(Sol))_{\Uparrow(I_1 \cup O_1)} \cap Init(IO)^{*}{\Uparrow(U \cup V)}$ существует продолжение в языке $(L(C)_{\Uparrow I2 \cup O2} \cap (L(Sol)_{\Uparrow I1 \cup O1} \cap (IO)^{*}{\Uparrow(U \cup V)}$. Таким образом, решение Sol называется прогрессивным, если из каждого состояния полуавтомата $C_{\Uparrow(I_2 \cup O_2)} \cap Sol_{\Uparrow(I_1 \cup O_1)} \cap (IO)^{*}$ достижимо финальное состояние, и в каждом финальном состоянии существует переход по каждому символу из алфавита I.

2. Алгоритм нахождения наибольшего прогрессивного решения

Постановка задачи. Даны полностью определенные автоматы $C = (C, I_1 \cup V, O_1 \cup U, T_C, c_0)$ и $S = (S, I_1 \cup I_2, O_1 \cup O_2, T_S, s_0)$ и разрешимое уравнение $C \Diamond X \cong S$. Необходимо найти наибольшую редукцию $Prog(Sol)$ наибольшего решения Sol уравнения, которая является прогрессивным решением. Если такой редукции не существует, то уравнение $C \Diamond X \cong S$ не имеет прогрессивных решений. Если такая редукция существует, то $Prog(Sol)$ есть наибольшее прогрессивное решение уравнения.

Алгоритм 1. Построение наибольшего прогрессивного решения (если прогрессивные решения существуют)

Вход: Полностью определенное наибольшее решение Sol уравнения $C \Diamond X \cong S$.

Выход: Наибольшая полностью определенная редукция $Prog(Sol)$ наибольшего решения, которая является прогрессивным решением (если прогрессивные решения существуют).

1. Положим $j := 1$ и $P^j = Sol$.
2. Построим полуавтомат $R^j = C_{\Uparrow(I_2 \cup O_2)} \cap P^j{}_{\Uparrow(I_1 \cup O_1)} \cap (IO)^{*}{\Uparrow(U \cup V)}$ и положим $R := R^j$.
3. Удалим в полуавтомате R все состояния (и переходы в эти состояния), в которых генерируется пустой язык. Если в полученном полуавтомате из каждого финального состояния есть переходы по всем внешним входным символам, то обозначаем полученный полуавтомат B^j и Шаг 5. Иначе Шаг 4.

4) Удаляем из B^j финальные состояния (и переходы в эти состояния), из которых нет перехода хотя бы для одного внешнего входного символа. Если удаляется начальное состояние, то прогрессивного решения не существует, КОНЕЦ. Иначе обозначаем полученный полуавтомат R и Шаг 3.

5) Находим проекцию полуавтомата $((Init(C)_{\Uparrow(I_2 \cup O_2)} \cap Init(P^j)_{\Uparrow(I_1 \cup O_1)} \cap Init(IO)^{*}{\Uparrow(U \cup V)}) \setminus Init(B^j))$ на алфавиты решения и вычитаем ее из полуавтомата P^j. Полученный полуавтомат представим автоматом P^{j+1}. Если автоматы P^{j+1} и P^j эквивалентны, то автомат P^j есть наибольшее прогрессивное решение, КОНЕЦ. Иначе в автомате P^{j+1} итеративно удалим состояния (и переходы в эти состояния), из которых нет перехода хотя бы для одного входного символа. Если удаляется начальное состояние, то прогрессивного решения не существует, КОНЕЦ. Иначе увеличиваем j на 1, и Шаг 2.

Теорема 1. Автомат, построенный по алгоритму 1, является наибольшим прогрессивным решением.

Доказательство. На Шаге 5 алгоритм заканчивает работу, если и только если из языка полуавтомата P^j не было удалено ни одной последовательности, т.е. P^j соответствует полностью определенному автомату, такому, что в полуавтомате $C_{\Uparrow(I_2 \cup O_2)} \cap P^j{}_{\Uparrow(I_1 \cup O_1)} \cap (IO)^{*}{\Uparrow(U \cup V)}$ отсутствуют состояния, из которых не достижимо финальное состояние, и из каждого финального состояния есть переходы по всем внешним входным символам, т.е. P^j есть прогрессивное решение.

Покажем, что для любого j, $j \geq 0$, язык L каждого прогрессивного решения уравнения содержится в языке P^{j+1} автомата P^{j+1}. По определению наибольшего решения, $L \subseteq P^1 = Sol$.

Предположим, что $L \subseteq P^j$ для некоторого $j \geq 1$. Покажем, что $L \subseteq P^{j+1}$. Так как $L \subseteq P^j$, то

$$R = Init(C)_{\Uparrow(I_2 \cup O_2)} \cap Init(L)_{\Uparrow(I_1 \cup O_1)} \cap Init(IO)^{*}{\Uparrow(U \cup V)} \subseteq Init(C)_{\Uparrow(I_2 \cup O_2)} \cap Init(P^j)_{\Uparrow(I_1 \cup O_1)} \cap Init(IO)^{*}{\Uparrow(U \cup V)}.$$

Пусть $Q^j = Init(C)_{\Uparrow(I_2 \cup O_2)} \cap Init(P^j)_{\Uparrow(I_1 \cup O_1)} \cap Init(IO)^{*}{\Uparrow(U \cup V)}$. Ниже представлены соотношения, которые показывают, что $L \cap (Q^j \setminus Init(B^j))_{\Downarrow(I_2 \cup O_2 \cup U \cup V)} = \varnothing$. Мы пользуемся равенством

$$(L_1 \cap L_2)_{\Downarrow(I_2 \cup U \cup V \cup O_2)} = L_1{}_{\Downarrow(I_2 \cup U \cup V \cup O_2)} \cap L_2{}_{\Downarrow(I_2 \cup U \cup V \cup O_2)},$$

так как язык $L_{\Uparrow(I_1 \cup O_1)}$ обладает свойством

$$((L_{\Uparrow(I_1 \cup O_1)})_{\Downarrow(I_2 \cup O_2 \cup U \cup V)})_{\Uparrow(I_1 \cup O_1)} = L_{\Uparrow(I_1 \cup O_1)}$$

и свойством

$$(L_{\Uparrow(I_1 \cup O_1)})_{\Downarrow(I_2 \cup O_2 \cup U \cup V)} = L \ [1].$$

Таким образом,

$R \subseteq Init(B^j) \subseteq Q^j \Rightarrow$

$R \cap (Q^j \setminus Init(B^j)) = \varnothing \Rightarrow$

$(Init(C)_{\Uparrow(I_2 \cup O_2)} \cap Init(L)_{\Uparrow(I_1 \cup O_1)}) \cap (Init(C)_{\Uparrow(I_2 \cup O_2)} \cap Init(P^j)_{\Uparrow(I_1 \cup O_1)} \cap Init(IO)^*{}_{\Uparrow(U \cup V)}) \cap \overline{Init(B^j)} = \varnothing \Rightarrow$

$Init(L)_{\Uparrow(I_1 \cup O_1)} \cap (Init(C)_{\Uparrow(I_2 \cup O_2)} \cap Init(P^j)_{\Uparrow(I_1 \cup O_1)} \cap Init(IO)^*{}_{\Uparrow(U \cup V)}) \cap \overline{Init(B^j)} = \varnothing \Rightarrow$

$L_{\Uparrow(I_1 \cup O_1)} \cap Q^j \cap \overline{Init(B^j)} = \varnothing \Rightarrow$

$L_{\Uparrow(I_1 \cup O_1)} \cap (Q^j \setminus Init(B^j)) = \varnothing.$

Так как $L \subseteq P^j$, $L \cap (Q^j \setminus Init(B^j))_{\Downarrow(I_2 \cup O_2 \cup U \cup V)} = \varnothing$ и $P^{j+1} = P^j \setminus (Q^j \setminus Init(B^j))_{\Downarrow(U \cup V \cup I_2 \cup O_2)}$, то $L \subseteq P^{j+1}$. ■

Теорема 2. Алгоритм 1 сходится.

Доказательство. Доказательство основывается на том факте, что язык каждого финального состояния полуавтомата R^j можно представить как конечное число объединений конечного числа пересечений языков (и их дополнений) состояний полуавтомата R^1, т.е. количество попарно не эквивалентных автоматов, т.е. автоматов с попарно различными языками, которое может быть сгенерировано на Шаге 5 алгоритма, является конечным. Поскольку на каждой итерации язык полуавтомата R^{j+1} строго содержится в языке полуавтомата R^j, то на некоторой итерации полуавтомат P^{j+1} будет эквивалентен полуавтомату P^j, т.е. алгоритм сходится.

Для простоты изложения мы только демонстрируем вышеизложенный факт на примере полуавтомата R^2, т.е. показываем, что язык каждого финального состояния R^2 может быть представлен как конечное число объединений конечного числа пересечений языков (и их дополнений) состояний полуавтомата R^1.

Рассмотрим языки $R^1 = C_{\Uparrow(I_2 \cup O_2)} \cap P^1{}_{\Uparrow(I_1 \cup O_1)}$ и $R^2 = C_{\Uparrow(I_2 \cup O_2)} \cap P^2{}_{\Uparrow(I_1 \cup O_1)}$, где C, P^1, P^2 есть соответственно языки полуавтоматов C, P^1, P^2. Пусть $Q^1 = Init(C)_{\Uparrow(I_2 \cup O_2)} \cap Init(P^1)_{\Uparrow(I_1 \cup O_1)} \cap Init(IO)^*{}_{\Uparrow(U \cup V)}$. Выразим R^2 через R^1:

$$\begin{aligned}
R^2 &= C_{\Uparrow(I_2 \cup O_2)} \cap P^2{}_{\Uparrow(I_1 \cup O_1)} = \\
&= C_{\Uparrow(I_2 \cup O_2)} \cap [P^1 \setminus (Q^1 \setminus Init(B^1))_{\Downarrow(I_2 \cup U \cup V \cup O_2)}]_{\Uparrow(I_1 \cup O_1)} = \\
&= C_{\Uparrow(I_2 \cup O_2)} \cap [P^1 \cap \overline{(Q^1 \setminus Init(B^1))_{\Downarrow(I_2 \cup U \cup V \cup O_2)}}]_{\Uparrow(I_1 \cup O_1)} = \\
&= C_{\Uparrow(I_2 \cup O_2)} \cap P^1{}_{\Uparrow(I_1 \cup O_1)} \cap (\overline{(Q^1 \setminus Init(B^1))_{\Downarrow(I_2 \cup U \cup V \cup O_2)}})_{\Uparrow(I_1 \cup O_1)} = \\
&= R^1 \cap (\overline{(Q^1 \setminus Init(B^1))_{\Downarrow(I_2 \cup U \cup V \cup O_2)}})_{\Uparrow(I_1 \cup O_1)}.
\end{aligned}$$

Заметим, что при построении R^2 мы вновь воспользовались равенством $(L_1 \cap L_2)_{\Uparrow(I_1 \cup O_1)} = L_1{}_{\Uparrow(I_1 \cup O_1)} \cap L_2{}_{\Uparrow(I_1 \cup O_1)}$.

Рассмотрим конструктивные шаги, ведущие от языка R^1 к языку R^2:

- Язык $(Q^1 \setminus Init(B^1))$ есть язык полуавтомата $M = (Init(\mathrm{C})_{\Uparrow(I_2 \cup O_2)} \cap Init(\mathrm{P}^1)_{\Uparrow(I_1 \cup O_1)} \cap Init(IO)^*{}_{\Uparrow(U \cup V)}) \setminus Init(B^1)$, каждое финальное состояние которого представляет язык $L(r^1)$, который представлен состоянием r^1 полуавтомата R^1.
- Язык $((Q^1 \setminus Init(B^1))_{\Downarrow(I_2 \cup U \cup V \cup O_2)})_{\Uparrow(I_1 \cup O_1)}$ есть язык полуавтомата $M_{\Downarrow(U \cup V \cup I_2 \cup O_2)}$, каждое финальное состояние которого представляет язык $L(r^1)_{\Downarrow(I_2 \cup U \cup V \cup O_2)}$.
- Язык $R^1 \cap (\overline{(Q^1 \setminus Init(B^1))_{\Downarrow(I_2 \cup U \cup V \cup O_2)}})_{\Uparrow(I_1 \cup O_1)}$ есть язык полуавтомата $R^1 \cap (\overline{M_{\Downarrow(U \cup V \cup I_2 \cup O_2)}})_{\Uparrow(I_1 \cup O_1)}$.

Для того чтобы найти дополнение $\overline{M_{\Downarrow(U \cup V \cup I_2 \cup O_2)}}$, необходимо детерминизировать полуавтомат $M_{\Downarrow(U \cup V \cup I_2 \cup O_2)}$. Каждое состояние $\widehat{m}^1_j = \{m^1_h \in M \mid h \in H^1_j \subseteq J^1\}$ детерминированного полуавтомата будет представлять язык $L(\widehat{m}^1_j) = \bigcap_{h \in H^1_j \subseteq J^1} L(m^1_h)_{\Downarrow(U \cup V \cup I_2 \cup O_2)} = \bigcap_{h \in H^1_j \subseteq J^1} L(r^1_h)_{\Downarrow(U \cup V \cup I_2 \cup O_2)}$, где $J^1 = \{1, ..., |M|\}$,

M – множество состояний полуавтомата $M_{\Downarrow(U\cup V\cup I_2\cup O_2)}$. Кроме того, к полуавтомату добавляется *безразличное* состояние $\{dc\}$, которое представляет язык $L(\{dc\})$, полученный дополнением объединения языков, представленных финальными состояниями детерминированного полуавтомата:

$$L(\{dc\}) = \overline{\bigcup_{j\in\hat{J}^1} L(\hat{m}_j^1)} = \bigcap_{j\in\hat{J}^1} \overline{L(\hat{m}_j^1)} = \bigcap_{j\in\hat{J}^1} \overline{\bigcap_{h\in H_j^1\subseteq J^1} L(r_h^1)_{\Downarrow(U\cup V\cup I_2\cup O_2)}} =$$
$$= \bigcap_{j\in\hat{J}^1} \bigcup_{h\in H_j^1\subseteq J^1} \overline{L(r_h^1)_{\Downarrow(U\cup V\cup I_2\cup O_2)}} = \bigcup_{K\in\mathcal{K}\subseteq 2^{J^1}} \bigcap_{k\in K} \overline{L(r_k^1)_{\Downarrow(U\cup V\cup I_2\cup O_2)}},$$

где $\hat{J}^1 = \{1,...,|\widehat{M}|\}$ и $\widehat{M}$ есть множество состояний детерминированного полуавтомата.

Полуавтомат $\overline{M_{\Downarrow(U\cup V\cup I_2\cup O_2)}}$ получается из детерминированного полуавтомата переключением финальных и нефинальных состояний. Каждое финальное состояние $r_j^2 = (r_j^1, \hat{m}_p^1)$ полуавтомата $R^2 = R^1 \cap (\overline{M_{\Downarrow(U\cup V\cup I_2\cup O_2)}})_{\Uparrow(I_1\cup O_1)}$ представляет язык $L(r_j^2)$, который можно выразить в следующей форме: если $\hat{m}_p^1 \neq \{dc\}$, то

$$L(r_j^2) = L(r_j^1) \cap \bigcap_{h\in H_j^1\subseteq J^1} (L(r_h^1)_{\Downarrow(U\cup V\cup I_2\cup O_2)})_{\Uparrow(I_1\cup O_1)};$$

если $\hat{m}_p^1 = \{dc\}$, то

$$L(r_j^2) = L(r_j^1) \cap [\bigcup_{K\in\mathcal{K}\subseteq 2^{J^1}} \bigcap_{k\in K} \overline{L(r_k^1)_{\Downarrow(U\cup V\cup I_2\cup O_2)}}_{\Uparrow(I_1\cup O_1)}] = \bigcup_{K\in\mathcal{K}\subseteq 2^{J^1}} [L(r_j^1) \cap \bigcap_{k\in K} \overline{L(r_k^1)_{\Downarrow(U\cup V\cup I_2\cup O_2)}}_{\Uparrow(I_1\cup O_1)}]. \blacksquare$$

Таким образом, если наибольшее прогрессивное решение существует, то уравнение имеет прогрессивное решение, и более того, любое прогрессивное решение есть редукция наибольшего прогрессивного решения. Однако заметим, что, как показывает следующий пример, в общем случае не любая полностью определенная редукция наибольшего прогрессивного решения есть решение уравнения.

Рассмотрим уравнение $C \Diamond X \cong S$ для автоматов, представленных на рис. 2, *a* и *b*. Автомат, соответствующий наибольшему прогрессивному решению *Sol*, показан на рис. 2, *c*. Непосредственной проверкой нетрудно убедиться, что полностью определенная редукция наибольшего прогрессивного решения, показанная на рис. 2, *f*, не является прогрессивным решением уравнения. Полуавтомат, соответствующий композиции наибольшего прогрессивного решения и контекста, показан на рис. 2, *d*; полуавтомат, соответствующий композиции контекста и полностью определенной редукции *Red* наибольшего прогрессивного решения, показан на рис. 2, *e*.

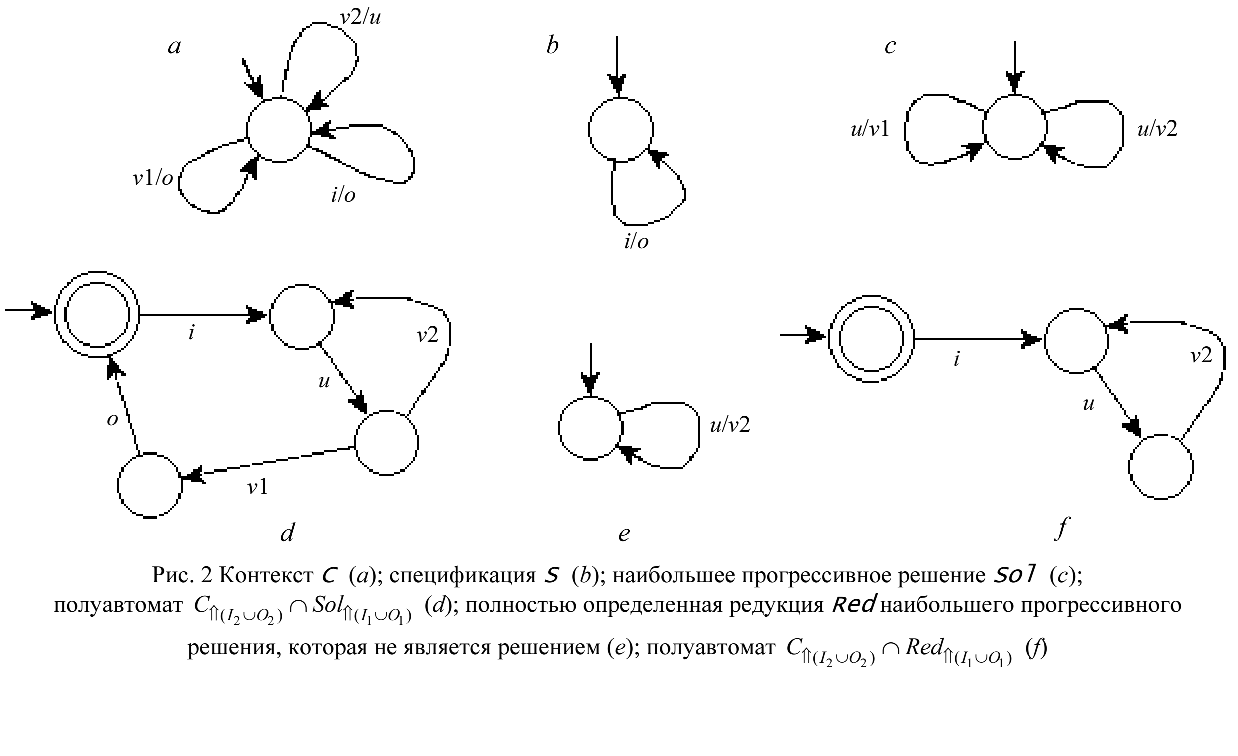

Рис. 2 Контекст *C* (*a*); спецификация *S* (*b*); наибольшее прогрессивное решение *Sol* (*c*); полуавтомат $C_{\Uparrow(I_2\cup O_2)} \cap Sol_{\Uparrow(I_1\cup O_1)}$ (*d*); полностью определенная редукция *Red* наибольшего прогрессивного решения, которая не является решением (*e*); полуавтомат $C_{\Uparrow(I_2\cup O_2)} \cap Red_{\Uparrow(I_1\cup O_1)}$ (*f*)

Заключение

В данной работе предложен алгоритм построения наибольшего прогрессивного решения параллельного автоматного уравнения, основанный на удалении из наибольшего решения последовательностей, которые ведут к возникновению заведомо тупиковых ситуаций в композиции с контекстом. Показана сходимость предложенного алгоритма. Показано также, что не каждая полностью определенная редукция наибольшего прогрессивного решения есть прогрессивное решение, и таким образом, проблема полной характеризации прогрессивных решений требует дополнительных исследований.

ЛИТЕРАТУРА

1. *Yevtushenko N.*, *Villa T.*, *Brayton R.K.*, *et al.* Sequential synthesis by language equation solving // International Workshop on Logic Synthesis. June, 2000.
2. *El-Fakih K.*, *Yevtushenko N.*, *Buffalov S.*, *Bochmann G.V.* Progressive solutions to a parallel automata equation // Theoretical Computer Science. October, 2006. P. 17 – 32.
3. *Yevtushenko N.*, *Villa T.*, *Brayton R.K.*, *et al.* Compositionally Progressive Solutions of Synchronous FSM Equation // Discrete Event Dynamic Systems. January, 2008. P. 51 – 89.