

Если измеряемый спектр содержит искажения, связанные с эффектом множественного рождения фотонов, то эти искажения останутся и в восстановленном по выше приведенной методике спектре. Чтобы получить истинный спектр фотонов этом случае, необходимо дополнительно применить процедуру преобразования спектра с использованием данных о распределении вероятности множественного рождения, которые могут быть получены либо в специальном эксперименте, подобном указанному выше, либо вычислены теоретически для конкретного случая взаимодействия релятивистской частицы с кристаллической мишенью.

Работа выполнена при поддержке Фонда Фундаментальных Исследований России грант № 00-02-17734

Библиографический список

10. Н.А.Булгаков и др. Препринт 1-84-630, Дубна, ОИЯИ, 1984
11. Eh.N.Tsyganov, Fiz.Elem.Chastits at Yadra 20(1989)15
12. М.Д.Бавижев, Ю В.Нильсон, Ю.А.Юрьев ЖЭТФ 95(1989) №4 1392-1397.
13. A.Belcasem et al., Phys. Rev.Lett. 54(1985)2667
14. И.Е.Внуков и др. Изв. ВУЗов.Физика №6 (1991) 88
15. S.V.Blazhevich, G.L.Bochek, I.E.Vnukov и др. // Proceedings of International symposium on radiation of relativistic electrons in periodical structures. September 6-10, 1993, Tomsk, Russia, s.176-181.
16. С.Ю.Амосов, В.Н.Каплин, Г.А.Науменко at al. // Proceedings of International symposium on radiation of relativistic electrons in periodical structures. September 6-10, 1993, Tomsk, Russia, s.185-193.

УДК 519.1

ТРАНСФОРМАЦИОННЫЕ МЕТОДЫ СРАВНЕНИЯ МОДЕЛЕЙ НА ЭКВИВАЛЕНТНОСТЬ

В.Е. Хачатрян

В теории программирования при изучении различных моделей программ [1-3], а значит при рассмотрении того или иного отношения эквивалентности между этими моделями, должны быть исследованы две наиболее важные задачи:

- разрешимость рассматриваемого отношения эквивалентности;
- разработка полной системы эквивалентных преобразований.

Положительное решение первой задачи очень важно, так как позволяет распознать эквивалентность для любой пары моделей.

Наличие эквивалентных преобразований позволяет рассматривать различные оптимизационные задачи. При решении этих задач обычно отыскивается некоторый градиент, использующий эквивалентные преобразования и позволяющий из исходной модели получить ей эквивалентную, обладающую более «хорошими» характеристиками.

Доказательство разрешимости отношений эквивалентности моделей программ обычно связывают с доказательством разрешимости проблемы включения. Поскольку:

$$A \sim B \leftrightarrow (A \subseteq B) \vee (B \subseteq A),$$

то из разрешимости проблемы включения автоматически следует разрешимость самой эквивалентности.

Существуют классы вычислительных моделей, в которых проблема эквивалентности разрешима, в то время как неразрешима проблема включения. Таковым является, например, класс многоленточных автоматов [4].

Нами предлагается методологически новый подход при построении алгоритмов разрешения отношений эквивалентности основанный на сравнении моделей и:

- не учитывающий проблему включения;
- использующий систему эквивалентный преобразований.

В работе рассматриваются алгоритмы, обладающие вышеуказанными свойствами для двух отношений эквивалентности.

Работа состоит из трех разделов. В первом разделе описываются рассматриваемые объекты, отношения между ними, приводятся некоторые общие понятия. Во втором приводятся алгоритмы распознавания первого из рассматриваемых отношений эквивалентности – строгой эквивалентности. В третьем исследуется второе, более слабое отношение эквивалентности.

1. Исследуемые объекты и отношения

Рассматриваемые модели, – детерминированные конечные автоматы Рабина и Скотта и многоленточные автоматы [4] - будут изображаться в виде диаграмм переходов [5] и называться *диаграммами*.

Диаграммы строятся над двумя конечными алфавитами:

$$P = \{p_1, p_2, \dots, p_n\}, n \geq 2,$$

$$Q = \{0, 1\}.$$

По определению, *диаграмма* – это конечный ориентированный граф с размеченными вершинами и дугами. Ее структура удовлетворяет следующим требованиям:

- в ней имеются две выделенные вершины, называемые входом и выходом диаграммы; из выхода нет исходящих дуг, а из всех остальных вершин исходят ровно две дуги;

- все вершины, кроме выхода, помечены символами алфавита P , а выходящие из вершин дуги помечены различными символами алфавита Q .

Для простоты иллюстраций во всех приводимых примерах $P = \{p, q\}$; вершины диаграммы, отличные от ее входа и выхода, изображаются полными кружочками; вход затушеванным, а выход - перечеркнутым кружочком; дуга, которой сопоставлено число 1, изображается жирной точкой у ее основания; метки дуг опускаются.

На рис. 2 – 9 приведены примеры диаграмм.

Пусть D диаграмма над P и Q . Конечный ориентированный путь w в диаграмме D задается последовательностью дуг. Он описывается *историей* $l(w)$, где

$$l(w) = (a_1, \varepsilon_1), (a_2, \varepsilon_2), \dots, (a_n, \varepsilon_n),$$

a_j - метка вершины, из которой исходит j -ая дуга пути, а ε_j - метка этой дуги, $j=1, 2, \dots, n$.

Путь w , начинающийся во входе диаграммы, назовем *маршрутом*; если последний оканчивается в выходе, то – *маршрутом через диаграмму*.

Диаграммы D_1, D_2 , по определению, *строго эквивалентны* ($D_1 \sim D_2$) тогда и только тогда, когда для любого маршрута через $D_j, j=1, 2$, существует маршрут через D_{3-j} , имеющий ту же историю.

p_i -*проекцией*, $i=1, 2, \dots, n$, пути w , называется слово, полученное из $l(w)$ удалением всех пар, не содержащих символа p_i .

Диаграммы D_1, D_2 , по определению, *эквивалентны* ($D_1 \approx D_2$) тогда и только тогда, когда для любого маршрута через $D_j, j=1, 2$, существует маршрут через D_{3-j} , все p_i -проекции которого $i=1, 2, \dots, n$, совпадают с p_i -проекциями первого маршрута.

Очевидно, что $D_1 \sim D_2 \rightarrow D_1 \approx D_2$.

Формально преобразования диаграмм определяются заданием взаимозаменяемых фрагментов.

Фрагмент диаграммы - это ее часть, задаваемая множеством V вершин диаграммы и содержащая вместе с этими вершинами все инцидентные им дуги. Вершины из V и инцидентные им дуги сохраняют приписанные им в диаграмме метки. Вершины множества V именуется *внутренними вершинами фрагмента*. Дуга фрагмента, начинающаяся не во внутренней вершине, называется входящей, ее начало и конец – *внешними* и, соответственно, *внутренними входами* фрагмента. Дуга фрагмента, которая ведет не на внутреннюю вершину, называется *выходящей*, ее начало и конец – *выходами* фрагмента, *внутренним и внешним* соответственно.

Изоморфными называются фрагменты, отличающиеся не более, чем наименованиями, как внутренних вершин, так и их внешних входов и выходов.

Преобразование диаграмм осуществляется путем замены в диаграмме какого-либо ее фрагмента другим фрагментом. Такие пары взаимозаменяемых фрагментов и определяют систему преобразований, которая обычно задается набором аксиом [6].

Макровершиной диаграммы назовем максимальный ее фрагмент, обладающий свойством: для любой пары внутренних вершин этого фрагмента существует ориентированный путь, их соединяющий.

В ходе изложения договоримся рассматривать лишь *приведенные* диаграммы, которые являются *свободными*. Приведенные - это такие диаграммы, в которых всякая вершина диаграммы, кроме, быть может, единственного пустого цикла, принадлежит некоторому маршруту через нее. *Пустым циклом* называется фрагмент, содержащий лишь одну вершину, выходящие дуги из которой направлены в нее же.

Свободной диаграммой считается такая, в которой любая *макровершина* обладает единственной входящей дугой.

Заметим, что рассматриваемые нами отношения эквивалентности позволяют использовать лишь приведенные диаграммы, которые являются свободными, поскольку в противном случае, с помощью соответствующих преобразований их можно сделать таковыми [7]. Эти преобразования сохраняют рассматриваемые нами отношения эквивалентности. Поэтому, используя любую диаграмму, мы будем считать ее приведенной и свободной, опуская рассмотрение тех дополнительных преобразований, которые делают ее таковой.

2. Распознавание строгой эквивалентности

Конечные автоматы в нашей работе представлены в виде диаграмм для которых определена строгая эквивалентность.

Напомним, что строгая эквивалентность диаграмм требует совпадения множеств историй путей, проходящих через них.

Для описания двух алгоритмов разрешения имеющих в своей основе различные подходы приведем некоторые дополнительные понятия.

Древовидное начало диаграммы D , обозначим его F_D - это фрагмент имеющий вид дерева. Входом его является вход диаграммы, а листьями являются входы в макровершины или выход диаграммы. Заметим, что древовидное начало может состоять и из одной вершины, если вход диаграммы принадлежит циклу.

Через $\tau_2(D_1, D_2)$ обозначим процедуру, которая по древовидному началу F_{D_1} диаграммы D_1 в диаграмме D_2 «выделяет» фрагмент F , изоморфный F_{D_1} , и выдает значение «да», если его можно построить, и «нет», в противном случае. Выделение изоморфного фрагмента зависит от рассматриваемого отношения эквивалентности. Для строгой эквивалентности достаточно воспользоваться преобразованиями типа «склейка-расклейка» [1].

Сечением S древовидного фрагмента F назовем множество таких его вершин $S = \{s_1, s_2, \dots, s_k\}$, которые удовлетворяют условию: на каждой ветви начинающейся во входе фрагмента F , находится одна и только одна, вершина из S .

Пусть F, F^1 - древовидные фрагменты. Будем говорить, что F строго содержится в F^1 ($F \subset F^1$), если в F^1 существует сечение S , такое, что древовидный подфрагмент F^* фрагмента F , с тем же входом, что и F^1 и с выходами из S , изоморфен F ; все вершины из S являются либо внутренними вершинами фрагмента F , либо его выходом. Причем множество внутренних вершин сечения S не пусто.

Считается, что F содержится в F^1 ($F \subseteq F^1$), если, либо все вершины сечения S являются выходами фрагмента F^1 , либо существует вершина сечения S , не являющаяся ни выходом фрагмента F , ни внутренней вершиной того же фрагмента F .

Монотонной раскруткой диаграммы D назовем следующую модификацию D :

для каждого выхода древовидного начала F_D , являющегося входом макровершины, создается копия, на которую переносится выходящая дуга фрагмента F_D , а дуги, выходящие из копии, направляются по дугам образа. Напомним, что полученные диаграммы затем непременно превращаются в приведенные и свободные. Обозначим полученную диаграмму через $\tau_0(D)$.

Утверждение 1. Если диаграмма D содержит цикл, то $F_D \subset F_{\tau_0(D)}$.

Обозначим через F_D^∞ - древовидный фрагмент, содержащий все пути через диаграмму D . Легко заметить, что если D содержит цикл, то это дерево бесконечно.

Пусть $\tau_0^0(D) = D$, $\tau_0^{i+1}(D) = \tau_0(\tau_0^i(D))$.

Утверждение 2. $\lim_{i \rightarrow \infty} F_{\tau_0^i(D)} = F_D^\infty$.

Справедливость утверждения следует из возможности увеличения древовидного фрагмента по всем направлениям продвижения.

Пусть $F \equiv F_{D_1}$ с сечением $S_1 = \{s_1^1, \dots, s_k^1\}$ и $F \equiv F_{D_2}$ с сечением $S_2 = \{s_1^2, \dots, s_k^2\}$. Здесь и в дальнейшем символом \equiv обозначается изоморфизм фрагментов. Отметим также, что диаграммы являются фрагментами.

Справедлива

Теорема 1. $D_1 \sim D_2 \leftrightarrow (\forall_{i=1, \dots, k})(D_1(s_i^1) \sim D_2(s_i^2))$.

Для распознавания строгой эквивалентности предлагается следующее достаточное условие

Лемма 1. Пусть D_1, D_2 - диаграммы и $D_1^* = \tau_0^i(D_1)$, $i > 0$, $D_2 = \tau_2(\tau_0^i(D_1), D_2)$; $S_1 = \{s_1^1, \dots, s_k^1\}$ - сечение задающее древовидное начало F_{D_1} , а $S_2 = \{s_1^2, \dots, s_k^2\}$ - сечение задающее изоморфный ему фрагмент в D_2 ; $I_k = \{1, 2, \dots, k\}$, $I' \subseteq I_k$; $(\forall_i)(i \in I')(D_1(s_i^1) \sim D_2(s_i^2))$.

Тогда для строгой эквивалентности диаграмм D_1, D_2 ($D_1 \sim D_2$) достаточно, чтобы

$$(\forall_i)(i \in (I' \setminus I_k))(((D_1^*(s_i^1), D_2^*(s_i^2))) \in \{D_1, D_2\}) \vee (D_1^*(s_i^1) \equiv D_2^*(s_i^2)).$$

Доказательство Рассмотрим наиболее сложный случай, когда поддиаграммы $D_1(s_i^1), D_2(s_i^2)$ принадлежат множеству $\{D_1, D_2\}$ и не совпадают. Для определенности положим, что $D_2(s_i^2) \equiv D_1$, а $D_1(s_i^1) \equiv D_2$ (остальные случаи рассматриваются аналогично).

Заменим в диаграммах D_1^* и D_2^* встретившиеся в них поддиаграммы D_1 и D_2 на строго эквивалентные им диаграммы $D_1 = \tau_0^i(D_1)$ и $D_2 = \tau_2(\tau_0^i(D_1), D_2)$. Обозначим полученные диаграммы через D_1^1 и D_2^2 соответственно.

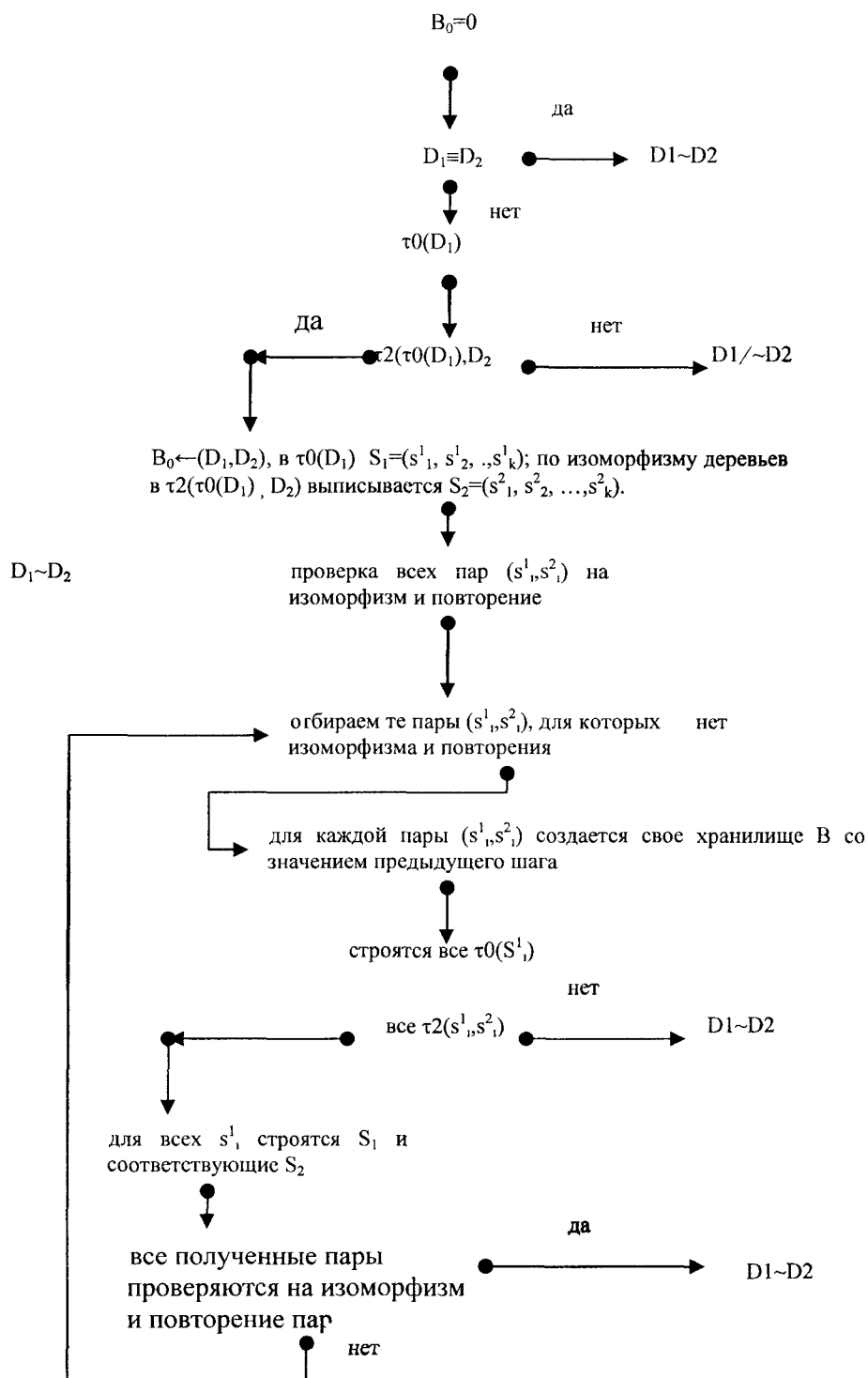
Тогда $D_1 \sim D_1^* \sim D_1^1, D_2 \sim D_2^* \sim D_2^2$ и

1. $F_{D_1} \subset F_{D_1^1}$;
2. процедура $\tau_2(D_1^1, D_2^2)$ - определена.

Этот процесс можно продолжить, поскольку повторение встреч с парой диаграмм D_1, D_2 продолжится $(D_1^j, D_2^j), j=1,2,\dots$

Первое условие означает, что для любого N , путь длины N через диаграмму D_1 , для некоторого j будет путем фрагмента $F_{D_1^j}$ и будет заканчиваться в его выходе.

Второе условие означает, что все пути через диаграмму D_1 длины не более N являются также путями длины не более N и через диаграмму D_2 и только они.



А это и означает, что $D_1 \sim D_2$.

На рис. 1 приведен алгоритм ρ_1 , в основе которого лежит процедура τ_2 . Алгоритм ρ_1 выявляет, строго эквивалентна или нет пара диаграмм, поданная на его вход.

Он сравнивает каждую пару поддиаграмм, возникающую в ходе ее выполнения. Сравнение считается удачным, если сравниваемая пара либо изоморфна, либо уже встречалась ранее. Для возможности проверки второго используются хранилища пройденных пар диаграмм. Они обозначаются через V . Заметим, что для каждого направления разворачивающего дерева используется свое хранилище. Кроме того, факт повторения встретившейся пары фиксируется сразу для всех направлений продвижения.

Теорема 2. Процедура ρ_1 является алгоритмом разрешения строгой эквивалентности.

Доказательство. Из утверждения 2 следует, что множество сравниваемых путей через диаграммы D_1, D_2 (по длине) «монотонно» возрастает. Это означает, что если $D_1 \sim D_2$, то процедура обязательно выдаст ответ «нет».

В случае же, когда $D_1 \sim D_2$, процедура ρ_1 будет выходить либо на изоморфизм проверяемых пар, либо на их повторение. Это связано с тем, что сравниваются лишь поддиаграммы диаграмм D_1, D_2 , а их конечное множество. Значит, выполняется условие достаточного признака эквивалентности (лемма 1), и процедура ρ_1 выдаст ответ «да», что означает $D_1 \sim D_2$.

Пример использования алгоритма ρ_1 приведен на рис. 2.

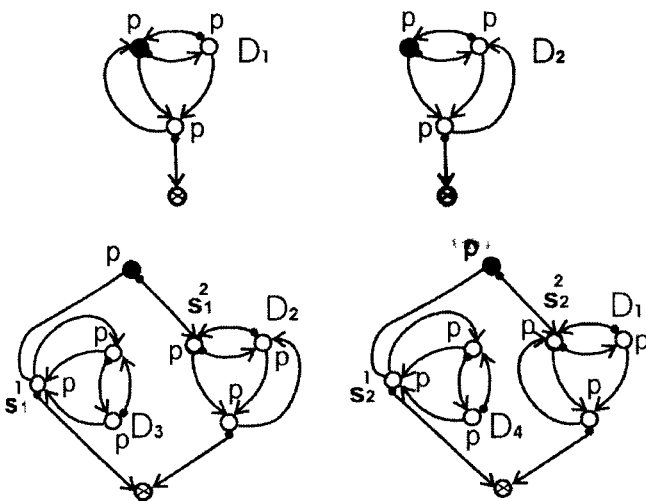


РИС. 2.

Здесь через D_1, D_2 обозначены исходные диаграммы; они не изоморфны; $F_{D_1} = \emptyset$, поэтому по D_1 строится диаграмма $\tau_0(D_1)$, а по D_2 и $\tau_0(D_1)$ - диаграмма $\tau_2(\tau_0(D_1), D_2)$; $S_1 = \{s^1_1, s^1_2\}$, а $S_2 = \{s^2_1, s^2_2\}$. Поскольку $D_3 \equiv D_4$, а пара D_2, D_1 повторилась, то $D_1 \sim D_2$.

Заметим, что алгоритм ρ_1 «удобно» использовать в тех случаях, когда «достаточно наполнена» база необходимыми и уже проверенными на эквивалентность парами диаграмм.

Для рассмотрения второго алгоритма приведем некоторые дополнительные понятия.

Определим характеристику диаграммы D - *древовидный фрагмент диаграммы* F_D . Все вершины F_D являются образами вершин диаграммы D . Корнем является образ входа диаграммы D . Обозначим через S список всех вершин диаграммы D , за исключением его выходов. Внося в F_D какую-либо вершину, за исключением выхода, будем вычеркивать ее образ из S .

На первом шаге в F_D вносится корень - образ входа v_0 диаграммы D , и вершина v_0 удаляется из списка S . Пусть на некотором шаге в F_D внесена вершина v , являющаяся образом вершины v диаграммы D , и вершина v вычеркнута из S . Пусть α_1 и α_2 - дуги, исходящие из вершины v и оканчивающиеся в вершинах v_1 и v_2 соответственно. Если v_i ($i=1,2$) содержится в S , создаем образ вершины v_i , направляем в него дугу α_i и удаляем v_i из списка S . Если v_i не содержится в списке S , но не является выходом, то создаем образ вершины v_i , объявляем его *промежуточным выходом* и в него направляем дугу α_i . Наконец, если v_i является выходом диаграммы D , то создаем образ v_i , он объявляется *выходом* для F_D . Дугу α_i направляем в этот выход.

Отметим, что древовидный фрагмент, порожденный диаграммой, строится неоднозначно. Пример диаграммы и ее древовидных фрагментов приведен на рис.3.

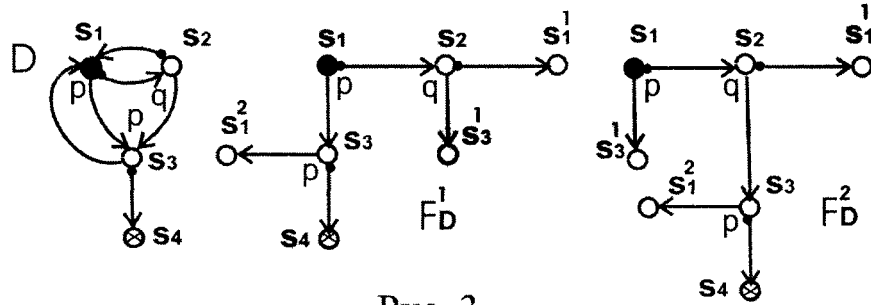


Рис. 3.

Покрытием диаграммы D назовем пару, состоящую из древовидного фрагмента F_D и множества S - всех пар вида (u, u') , где u, u' - вершины фрагмента F_D , порожденные общей вершиной диаграммы D , не являющейся выходом.

Через $D(v)$ обозначим поддиаграмму диаграммы D , входом которой является вершина v .

Под *разверткой* диаграммы D понимается диаграмма, обозначим ее $\sigma_1(D)$, состоящая из древовидного фрагмента F_D , каждый промежуточный выход u' которого заменён на поддиаграмму $D(u)$, где пара (u, u') - из S .

Пусть D_1, D_2 - диаграммы. Опишем *операцию выделения* древовидного фрагмента F_{D_1} в диаграмме D_2 . Для этого опишем процесс путешествия по диаграмме D_2 , использующий структуру древовидного фрагмента F_{D_1} .

Пусть u_0, w_0 входы, соответственно, F_{D_1} и D_2 . По меткам, установим взаимно однозначное соответствие между дугами, выходящими из этих вершин. Если такого соответствия установить нельзя, то операцию выделения считаем невыполнимой.

Рассмотрим пару u, w - вершины, в которые направлены соответствующие дуги в F_{D_1} и D_2 , выходящие из u_0, w_0 .

Возможны случаи:

- одна из вершин пары u, w является выходом, а другая - нет. Операция выделения в этом случае считается невыполнимой;
- обе вершины пары u, w не являются выходами. Тогда с этой парой поступаем подобно паре u_0, w_0 ;
- обе вершины пары u, w являются выходами. Тогда переходим к рассмотрению других имеющихся пар.

Замечим, что в нашем рассмотрении промежуточный выход в F_{D_1} не считается выходом.

Описанный процесс всегда конечен и, если во время путешествия мы не встретили невыполнимость, то строится диаграмма $\tau_2(\sigma_1(D_1), D_2)$ - результат операции выделения.

Обозначим $\sigma_1^0(D)=D; \sigma_1^i(D)=\sigma_1(\sigma_1^{i-1}(D))$.

Утверждение 3. $F(\sigma_1^i(D)) \subset F(\sigma_1^{i-1}(D)), i=1,2,\dots$

Утверждение 4. $\lim F(\sigma_1^i(D))=F^\infty_D$ при $i \rightarrow \infty$.

Пусть D_1, D_2 - диаграммы; $D_1'=\sigma_1(D_1); D_2'=\tau_2(\sigma_1(D_1), D_2)$; в D_1'

$S = \{(s_i, s_i'), s_i \sim s_i', i=1, \dots, n\}$; (d_i, d_i') - образы пар (s_i, s_i') в D_2' . Справедлива

Теорема 3. $D_1 \sim D_2 \leftrightarrow$ существует $\tau_2(\sigma_1(D_1), D_2)$ и $(\forall i)(D_2'(d_i) \sim D_2'(d_i')), i=1, \dots, n$.

Заменим в D_1' , для $i=1, \dots, n$, $D_1'(s_i')$ на $D_2'(d_i)$, или, что тоже самое, $D_2'(d_i')$ на $D_2'(d_i)$ и обозначим полученную диаграмму через $D_1'^*$.

Утверждение 5.

- $D_1 \sim D_2 \leftrightarrow D_1'^* \sim D_2'$;

- $D_1/\sim D_2 \leftrightarrow (D_1' \sim D_2') \& (D_1' \sim D_1)$;
- $(D_1/\sim D_2) \& (D_1' \sim D_2') \& (D_1/\sim D_1') \& (D_2 \sim D_2')$.

если $D_1 \sim D_2$, то во время преобразований класс эквивалентности для всех диаграмм один и тот же;

если $D_1/\sim D_2$, то у первой диаграммы класс эквивалентности меняется и при этом сохраняется не эквивалентность со второй диаграммой. Вторая диаграмма преобразуется в эквивалентную;

обозначим через D_2^{1*} диаграмму в которую преобразуется D_2^1 , если в ней заменить все поддиаграммы $D_2'(d_i')$ на $\tau_2(D_2'(d_i), D_2'(d_i'))$. Будем считать, что этот процесс не нарушается, т.е. его можно продолжить $(D_1, D_2) \rightarrow (D_1', D_2') \rightarrow (D_1^2, D_2^2) \rightarrow \dots (D_1^i, D_2^i)$.

Для пары диаграмм D_1, D_2 введём понятие «близости». Будем считать, что пара диаграмм имеет такую меру $h(D_1, D_2)$, если $F_{D_1} \subseteq F_{D_2}$, т.е. $F_{D_1} \cap F_{D_2} = F_{D_1}$.

Будем говорить, что пара D_1', D_2' обладает большей близостью, чем пара D_1, D_2 и обозначать $h(D_1', D_2') < h(D_1, D_2)$, если $F_{D_1} \subset F_{D_2}$. Пусть $D_1^0 = D_1, D_2^0 = D_2$.

Утверждение 6.

Если алгоритм ρ не ломается, то

- $i < j \rightarrow h(D_1^i, D_2^i) \leq h(D_1^j, D_2^j), i=0,1,\dots$
- если $h(D_1^i, D_2^i) = h(D_1^{i+1}, D_2^{i+1})$, то для $j < i$ $h(D_1^j, D_2^j) = h(D_1^i, D_2^i)$
- $\lim D_1^i = \lim D_2^i = F_D^\infty$, при $i \rightarrow \infty$, для некоторого D .

Пару (D_1, D_2) назовём *соизмеримой*, если алгоритм ρ применим к этой паре и завершается только парами (r, r') , для которых выполняется подтверждение эквивалентности за счёт изоморфизма, т.е. $\sigma_1(r) \equiv \tau_2(r, r')$.

Пару (d, d') назовём *повторяющейся*, если алгоритм ρ применим к этой паре, и завершается либо парами эквивалентность которых подтверждается изоморфизмом, либо повторяющейся парой (d, d') , либо некоторой другой повторяющейся парой.

Утверждение 7.

- Если пара (d, d') соизмеримая, то $d \sim d'$;
- Если пара (d, d') повторяющаяся, то $d \sim d'$;

Следствие утверждения 7. Для эквивалентности пары (d, d') достаточно, чтобы она была соизмеримой или повторяющейся.

Лемма 2. Если $\rho(D_1, D_2)$ не ломается, то множество пар сравниваемых поддиаграмм встречающихся в алгоритме ρ для заданной пары диаграмм D_1, D_2 - конечно.

Доказательство. Если алгоритм выдаёт «да» или «нет», то количество используемых пар диаграмм конечно в силу конечности самого процесса.

Рассмотрим случай, когда алгоритм ρ работает сколь угодно долго.

Обозначим через W множество пар диаграмм, участвующих в алгоритме ρ . Этому множеству принадлежит исходная пара (D_1, D_2) .

На первом шаге сравниваются диаграммы $D_1' = \sigma_1(D_1)$ и $\tau_2(D_1', D_2)$. Пусть $D = \tau_2(D_1', D_2)$; V - множество всех вершин диаграммы D .

W - содержит все пары $(D(d_i), D(d_i'))$, $i=1, \dots, n$, где (d_i, d_i') - образы пар эквивалентных вершин $(s_i, s_i') \in S$ в диаграмме $\sigma_1(D_1)$.

Назовём высотой древовидного фрагмента F , обозначим её через $|F|$, наибольшую длину его ветви. Все листья древовидного начала F_D являются входами в

поддиаграммы диаграммы D_2 , а значит и все древовидные начала диаграмм $D(di)$, $D(di')$, имеет высоту $\leq |F_{D_2}|$.

Пусть $D(v)$ - поддиаграмма D , со входом $v \in V$; $F(D(v))$ - древовидное покрытие $D(v)$; $|F(D(v))|$ - его высота и $N = \max |F(D(v))|$, $v \in V$.

На общем шаге алгоритма ρ пары $D(d_i)$, $D(d_i')$, в силу незавершённости алгоритма, подвергаются изменению, создаются $\sigma_1(D(d_i))$ и $\tau_2(D(d_i), D_2(d_i'))$, причем получаемые диаграммы не изоморфны.

Процедура σ_1 , примененная к $D(d_i)$, разворачивает поддиаграммы $D(d_{i1}')$, ..., $D(d_{im}')$, повисшие на листьях поддерева, начинающего в d_i , а значит $|F_{D(d_{ij}')}| \leq N$. Обозначим через $S^* = \{(s_e, s_e'), e=1, \dots, k\}$ пары эквивалентных вершин в $\sigma_1(D(d_i))$ и через (r_e, r_e') - соответствующие им пары в $\tau_2(D(d_i), D_2(d_i'))$. Выше указанному неравенству будут удовлетворять и поддерева в $\sigma_1(D(d_i))$ начинающиеся в s_e' и заканчивающиеся в s_{j1}' , ..., s_{jm}' . Но тогда этому неравенству будут удовлетворять и изоморфные им деревья в диаграмме $D^* = \tau_2(D(d_i), D_2(d_i'))$. Они начинаются в вершине r_e и заканчиваются в вершинах r_{j1}' , ..., r_{jm}' . Получаем, что поддиаграммы $D^*(r_e)$, $D^*(r_e')$ таковы, что $D^*(r_e)$ начинается древовидным фрагментом с высотой $\leq N$, на листьях которого висят поддиаграммы диаграммы D_2 , а $D^*(r_e')$ - просто является поддиаграммой D_2 . Поскольку сказанное верно и для всех пар диаграмм, полученных на последующих шагах, заключаем, что W - конечно.

Теорема 4. Процедура ρ является алгоритмом разрешения для строгой эквивалентности.

Доказательство. Если процедура ρ ломается, то $D_1 \not\sim D_2$. Если же не ломается, то ввиду справедливости леммы 2, применив достаточный признак эквивалентности диаграмм (следствие утверждения 7) получим, что $D_1 \sim D_2$.

На рис.4. приведен пример диаграмм, для которых применение алгоритма ρ приводит к паре диаграмм, которая является повторяющейся.

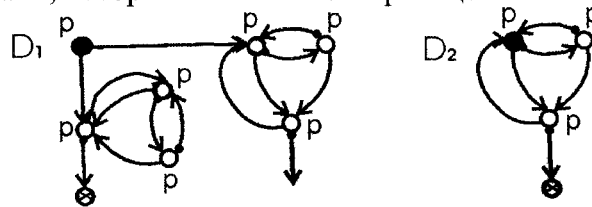


РИС. 4.

Покажем это. $D_1 \not\equiv D_2$. Обозначим $D_1' = \sigma_1(D_1)$ и $D_2' = \tau_2(D_1', D_2)$. На рис. 5. приведены диаграммы D_1' и D_2' . Легко видеть, что $D_1' \equiv D_2'$. Поэтому необходимо проверить на эквивалентность пары (d_1, d_1') , (d_1, d_1'') , (d_3, d_3') , (d_4, d_4') , (d_5, d_5') , (d_5, d_5'') . Легко проверить, что $\sigma_1(d_i) \equiv \tau_2(\sigma_1(d_i), d_i')$ для всех пар кроме (d_1, d_1'') .

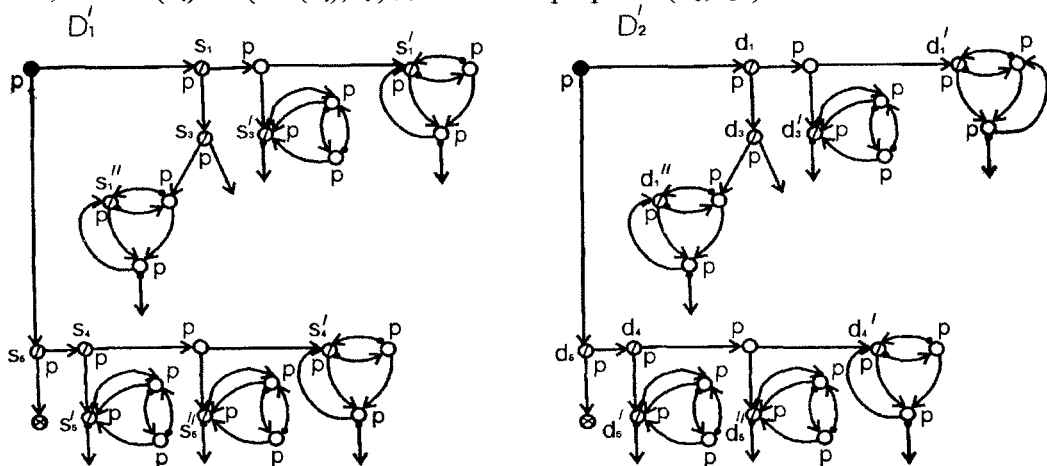


РИС. 5.

Поэтому рассмотрим лишь это продолжение. На рис.6. приведены диаграммы $D_1'(d_1)$ и $D_1'(d_1'')$, а на рис.7 и 8 - результат применения к ним процедур σ_1 и τ_2 . Не изоморфизм полученных диаграмм означает, что алгоритм необходимо продолжить. Т.е. необходимо проверить на эквивалентность пары (d_6, d_6') , (d_7, d_7') , (d_8, d_8') . Легко убедиться в том, что $\sigma_1(d_6) \equiv \tau_2(d_6, d_6')$ и $\sigma_1(d_8) \equiv \tau_2(d_8, d_8')$.

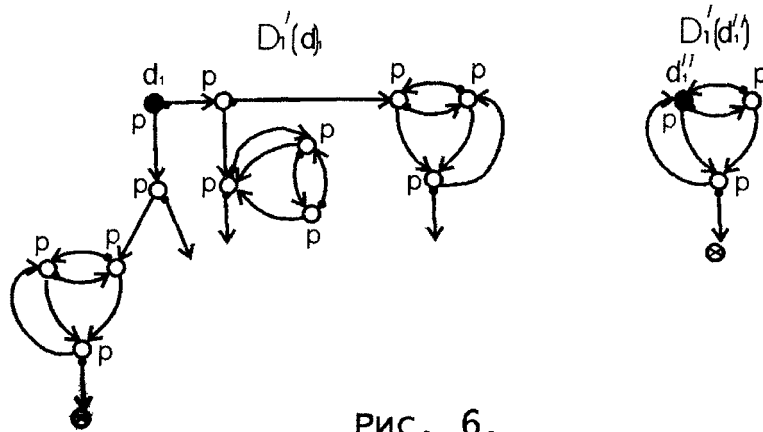


РИС. 6.

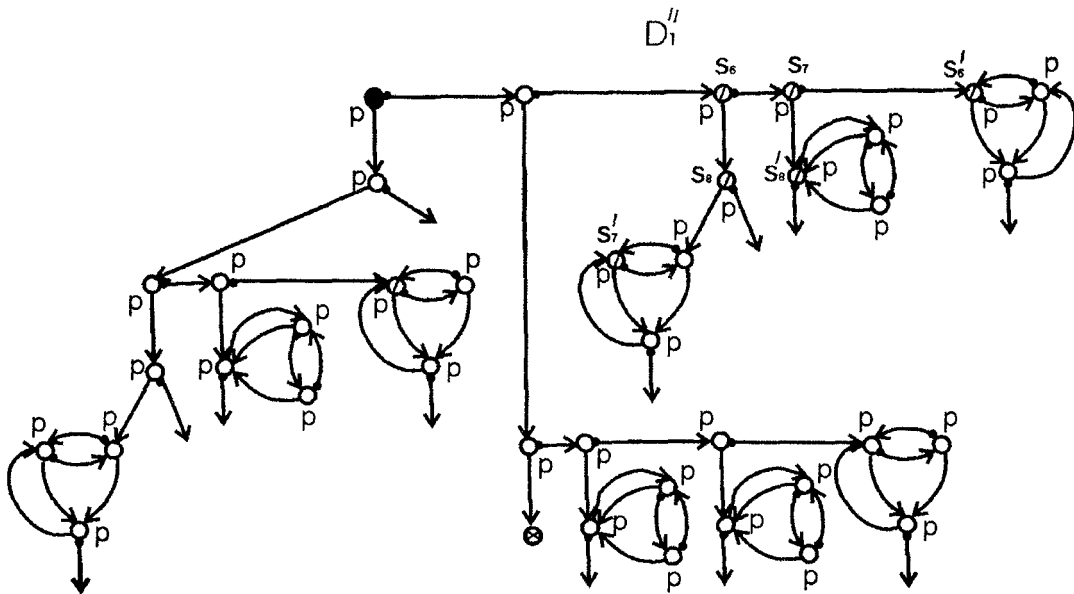


РИС. 7.

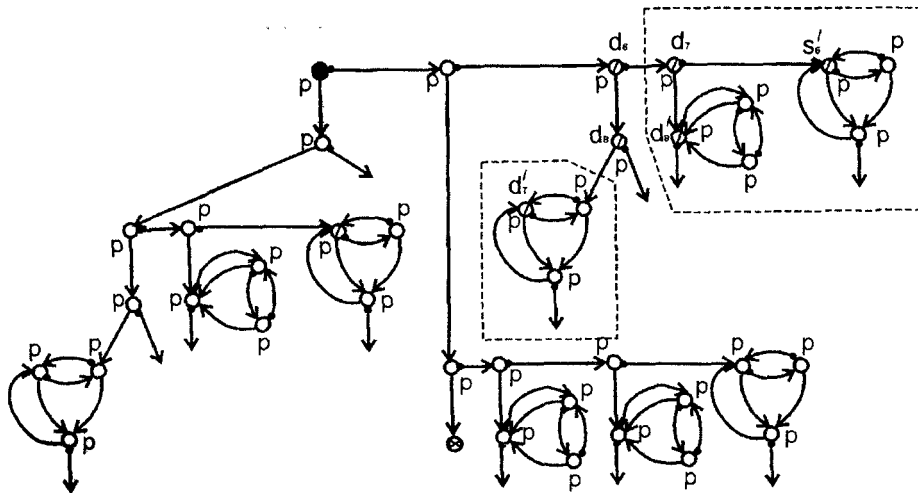


РИС. 8.

Пара же $\sigma_1(d_7), \tau_2(d_7, d_7')$ повторилась, поскольку она совпадает с парой исходных диаграмм (D_1, D_2) . На рис.8 эта пара выделена пунктирными линиями.

На рис.9 приведен пример диаграмм, обладающих свойством: $\rho(D_1, D_2)$ -завершается за счет изоморфизма и повторяемости; $\rho(D_2, D_1)$ - завершается только за счет изоморфизма.

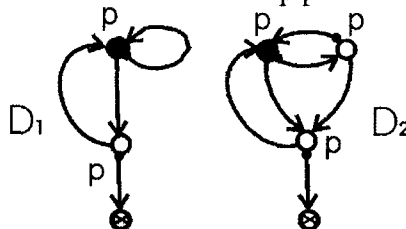


Рис. 9.

3. Распознавание эквивалентности

Алгоритмы структурного распознавания строгой (автоматной) эквивалентности могут быть обобщены, в частности, на эквивалентности, которые задаются перестановкой символов.

Предложенный выше алгоритм ρ модифицируется за счёт замены процедуры τ_2 на τ_2^* . В последней процедуре происходит выделение необходимой границы и перенос её в выравниваемое место древовидного фрагмента [8]. Обозначим полученный алгоритм через ρ^* . В процедуре τ_2^* используется преобразование [8], позволяющее менять порядок символов, используемых в диаграммах. Это осуществляется за счет применения преобразования, приведенного на рис.10. Оно сохраняет эквивалентность преобразуемых диаграмм.

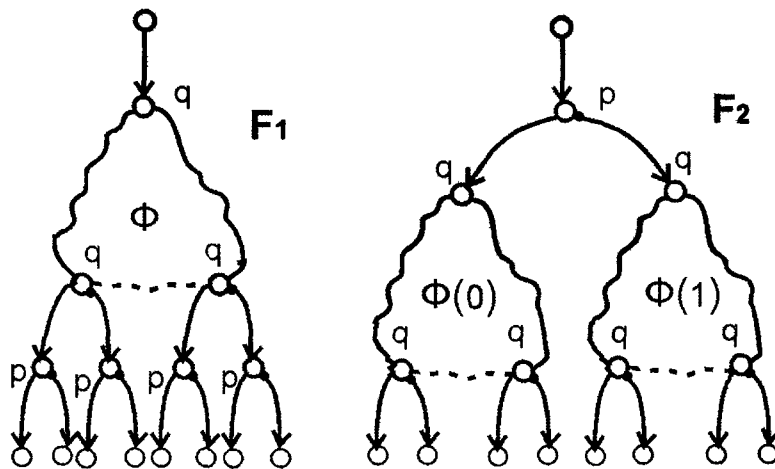


Рис. 10.

Эти алгоритмы являются алгоритмами разрешения в классе однородных [9] диаграмм в виду конечного числа пар сравниваемых на эквивалентность диаграмм, возникающих в ходе выполнения алгоритма. Диаграмма считается однородной, если вершины любого его цикла помечены общим для этого цикла символом.

Можно показать, что для заданной пары сравниваемых на эквивалентность диаграмм D_1, D_2 , множество пар диаграмм, создаваемых алгоритмом ρ^* - конечно.

Это означает, что предлагаемый алгоритм ρ^* является алгоритмом разрешения, поскольку возможны лишь следующие случаи:

-алгоритм ρ^* - ломается; $D_1 \not\sim D_2$;

-алгоритм ρ^* останавливается только за счет изоморфизма проверяемых на эквивалентность поддиаграмм (достаточное условие); $D_1 \sim D_2$;

-алгоритм ρ^* не останавливается, но появляются повторяющиеся пары. Этот факт фиксируется через определённое конечное число шагов (достаточное условие); $D_1 \sim D_2$

Докажем, что количество сравниваемых пар диаграмм, возникающих в алгоритме ρ^* конечно для класса однородных диаграмм.

Покажем, что для любой проверяемой пары (d, d') :

d' - является диаграммой некоторого конечного множества W ;

d - диаграмма, структура которой состоит:

- из древовидного фрагмента, высота которого ограничена для исходной пары диаграмм D_1, D_2 ;

- на листьях этого древовидного фрагмента расположены диаграммы из того же множества W

Макровершины диаграмм назовем *родственными*, если они, быть может, отличаются лишь входами.

Трассой диаграммы назовём упорядоченную последовательность вершин и макровершин этой макровершины, по которым можно из входа диаграммы попасть в выход;

Трассу А назовём *подобной трассе В*, если А можно получить из В, быть может, удалив из В некоторые вершины и макровершины и, быть может, заменив некоторые макровершины родственными.

Диаграмму D_1 , назовём *подобной* диаграмме D , если для каждой трассы из D_1 в диаграмме D найдётся ей подобная.

Под выносом из диаграммы D вершины данного типа будем понимать следующие преобразования диаграммы:

- найти ближайшие от входа D вершины данного типа;

- для найденных вершин, которые являются входами макровершин, создать копии; -

- направить выходящие из копий дуги в те поддиаграммы диаграмм D_1, D_2 , куда были направлены дуги их образов;

- перенести полученную границу во вход диаграммы, воспользовавшись заменой фрагмента F_1 на фрагмент F_2 (см. рис.10.).

Утверждение 8. Если D – однородная диаграмма, то поддиаграммы D_1, D_2 в которые направлены дуги вершины вынесенной из D – подобны D .

Утверждение 9. Множество диаграмм подобных D – конечно.

Обозначим через D_1', D_2' диаграммы, полученные после 1-го шага алгоритма ρ^* . Т.е. $D_1' = \sigma_1(D_1)$, $D_2' = \tau_2^*(D_1', D_2)$.

Через W – обозначим множество всех диаграмм, подобных поддиаграммам $D_2'(d_i')$, $i=1, \dots, k$. Обозначим через $N(D_1, D_2)$ максимальную высоту древовидных покрытий всех диаграмм из W .

Утверждение 10. Любая пара поддиаграмм алгоритма $\rho^*(D_1, D_2)$, задаваемая вершинами (r, r') , и предназначенная для сравнения, удовлетворяет условиям:

поддиаграмма с входом r' принадлежит W ;

поддиаграмма с входом r , начинается древовидным фрагментом с высотой не более N , листьями которого являются диаграммы из W .

Следствие утверждения 10 Число пар диаграмм, возникающих в алгоритме $\rho^*(D_1, D_2)$ и предназначенных для сравнения на эквивалентность, ограничено.

Доказательство утверждения основано на том, что при выделении из диаграммы вершины некоего типа, возникающие поддиаграммы подобны исходной

Теорема 5. Алгоритм ρ^* является алгоритмом разрешения эквивалентности в классе однородных диаграмм.

Библиографический список

1 Янов Ю.И. О логических схемах алгоритмов. //В сб.: Проблемы кибернетики.1958. вып 1, с.75-127.

2. Подловченко Р.И. О проблеме эквивалентных преобразований. //Программирование, 1986, № 6, с.3-14.
3. Подловченко Р.И. Система преобразований полная в классе схем программ с перестановочными операторами. //Программирование, 1998, № 2, с.58-67.
4. Rabin M.O. Scott D/ Finite automata and their decisions problems//IBM Journal of Research and Development. 1959. V. 3. № 2 P. 114-125. (Русский перевод: Кибернетический сборник. 1962. № 4. С.58-91).
5. Bird R. The equivalence problem for deterministic two tape automata // Journal of Computer and System Science. 1973. V. 7. № 4. P 218-236.
6. Подловченко Р.И., Айрапетян М.Г. О построении полных систем эквивалентных преобразований схем программ //Программирование, 1996, № 1, с.3-29.
7. Хачатрян В.Е. Полная система эквивалентных преобразований для многочленных автоматов //Программирование, 2003, № 1, с. 1-15.
8. Подловченко Р. И., Хачатрян В.Е. Об одном методологически новом алгоритме разрешения проблемы эквивалентности //Программирование, 2004, № 3, с. 1-17.
9. Хачатрян В.Е. Однородные логические графы //Прикладная математика. Ереван: Издательство Ереванского гос. Университета, 1981. с. 67-80.

СИСТЕМНЫЙ АНАЛИЗ И УПРАВЛЕНИЕ

УДК 681.51

РАВНОВЕСНЫЕ МАТЕМАТИЧЕСКИЕ МОДЕЛИ ДИНАМИЧЕСКИХ ПРОЦЕССОВ ЭЛЕКТРОПРИВОДА ПЕРЕМЕННОГО ТОКА

В.Д. Дмитриенко, Н.И. Заполовский, А.И. Баленко, В.И. Носков

В работах [1, 2] рассмотрены математические модели оптимальных систем управления, получающиеся при синтезе регуляторов методом аналитического конструирования по критерию обобщенной работы, когда объекты моделируются системами обыкновенных дифференциальных уравнений вида:

$$\frac{dx_i}{dt} + f_i(x_1, \dots, x_n, t) = \sum_{j=1}^m \varphi_{ij}(x_1, \dots, x_n, t) U_{ij} \sin(\alpha_j U_1 + \gamma_{ij}), \quad (1)$$

где $x_i = (\overline{i=1, n})$ - фазовые координаты объекта; $f_i, \varphi_{ij} (\overline{i=1, n}; \overline{j=1, m})$ - непрерывные функции; U_{ij}, U_1 - управляющие воздействия.

Показано, что оптимальными в смысле минимума функционала обобщенной работы

$$J = V_k[x_1(t_2), \dots, x_n(t_2), t_2] + \int_{t_1}^{t_2} Q(x_1, \dots, x_n, t) dt + \frac{1}{2} \int_{t_1}^{t_2} \frac{U_1^2 + U_{\text{лонт.}}^2}{k_1^2} dt + \quad (2)$$

$$+ \frac{1}{2} \int_{t_1}^{t_2} \sum_{i=1}^n \sum_{j=1}^m \frac{U_{ij}^2 + U_{ij\text{о.о.}}^2}{k_{ij}^2} dt$$