ОНТОЛОГИЧЕСКИЙ ВЗГЛЯД НА ТЕОРИЮ АВТОМАТОВ

Сергей Кривой, Людмила Матвеева, Е. Лукьянова, О. Седлецкая

***Аннотация**: Рассматривается краткая онтология теории автоматов, которая строится на принципах тип автомата – язык, распознаваемый этим автоматом, – приложения. Предлагаемая онтология не претендует на полноту, поскольку теория автоматов стала столь обширной, что обозреть все ее аспекты является трудной задачей.*

УДК 51.681.3

1. Введение

Одной из важных проблем в современном техническом и программном обеспечении компьютеров является проблема корректного проектирования и верификации полученного продукта. Если проблема проектирования относится к трудно формализуемым проблемам (а, следовательно, к трудно автоматизируемым), то проблема верификации является трудной в связи с тем, что она в себе аккумулирует множество разных задач из смежных областей, чем и объясняется ее сложность. Одной из таких смежных областей является теория автоматов, которая нашла и находит все новые и новые приложения в частичном решении проблем проектирования и верификации. В частности, если речь заходит о верификации свойств реактивных систем, то теория автоматов играет здесь ключевую роль, поскольку такие важные проблемы, как проблема распознавания свойств, достижимости определенного состояния или множества состояний и пустоты акцептируемого языка разрешимы в теории автоматов для большинства типов автоматов.

В данной работе построена краткая онтология теории автоматов исходя из таких зависимостей ***автомат – язык, распознаваемый этим автоматом, – приложения.*** В связи с ограниченным объемом данной работы, кратко рассматриваются только основные свойства автоматов и их приложения.

2. Конечные автоматы над конечными словами

Теория конечных автоматов над словами конечной длины является законченной теорией в том смысле, что все основные теоретические задачи этой теории решены. Основные результаты, которые получают в этой области, - это результаты прикладного характера, т. е. результаты применения методов теории автоматов к решению задач в конкретных прикладных областях. Однако, не смотря на это, теория конечных автоматов оказала влияние на дальнейшее развитие общей теории автоматов. Это проявляется в том, что появились многочисленные вариации понятия конечного автомата такие, как конечные автоматы над бесконечными словами [3, 12], временные автоматы [2], гибридные автоматы [9], автоматы над деревами [10] и т. д. и т. п.

Основным понятием, от которого строится онтология, является понятие конечного автомата над словами конечной длины в конечном алфавите. Среди таких автоматов наиболее употребительными являются три типа автоматов: *автоматы Мили, автоматы Мура и X –автоматы (или автоматы без выходов)*. Перейдем к определению этих типов автоматов.

Автоматы Мили. Пусть X = $\{x_1, x_2, \ldots, x_n\}$, Y = $\{y_1, y_2, \ldots, y_m\}$ – конечные алфавиты, т. е. конечные множества попарно различных элементов, которые называются символами или сигналами.

Определение 1. Пятерка *(A, X, Y, f, g)* называется автоматом Мили, если она состоит из множества *A* состояний автомата, алфавита входных символов *X*, алфавита выходных символов *Y,* функции переходов $f: A \times X \rightarrow A$ и функции· выходов $g: A \times X \rightarrow Y$. Алфавиты *X* и *Y* называют соответственно входным и выходным алфавитами символов или сигналов автомата.

Как правило, автомат обозначают символом множества его состояний, т. е. *A = (A, X, Y, f, g).* Если *f(a,x)* = *a'*, то говорят, что автомат *A* под действием входного сигнала $x \in X$ переходит в состояние *a'*, или что сигнал *x* переводит автомат *A* из состояния *a* в состояние *a'*. Если *g(a,x)* = *y*, то говорят, что автомат *A* преобразует в состоянии *a* входной сигнал $x \in X$ в выходной сигнал $y \in Y$. Функционирование автомата описывается таким образом. На вход автомата подается сигнал (символ) $x \in X$, под действием которого в соответствии с текущим состоянием автомата и функциями переходов и выходов изменяется состояние автомата и на его выходе появляется выходной сигнал (символ) $y \in Y$. Если на вход автомата подавать слово (последовательность входных символов), то на выходе также появляется слово (последовательность выходных символов). В этом случае автомат *A* можно рассматривать как алфавитный преобразователь информации, который отображает слова полугруппы *F(X)* в слова полугруппы *F(Y)*. Последовательность входных сигналов можно рассматривать как функцию натурального аргумента - дискретного автоматного времени. Это обстоятельство позволяет рассматривать автомат как дискретную динамическую систему, которая изменяет свои состояния во времени под действием внешних и внутренних факторов.

Автомат *A* называется инициальным, если в множестве его состояний выделено некоторое состояние a_0, которое называется начальным состоянием. При функционировании инициального автомата считается, что в начальный момент времени (перед подачей на его вход некоторого слова $p \in F(X)$) автомат находится в начальном состоянии a_0.

Автомат *A = (A, X, Y, f, g)* называется конечным, если все три множества *A, X* и *Y* конечны и бесконечным, если хотя бы одно из этих множеств бесконечно.

Автомат называется полным или полностью определенным, если функция переходов и выходов полностью определены и частичным, если хотя бы одна из этих функций частичная.

Автоматы у которых компонента *f* не является функцией, а является некоторым отношением, т. е. в таких автоматах не выполняется условие однозначности перехода, называются недетерминированными. Если же отношение *f* есть функцией, то автомат называется детерминированным. Следовательно, для детерминированного автомата *A* с начальным состоянием a_0 однозначно находится состояние *b*, в которое автомат переходит под действием слова $p \in F(X)$. А в недетерминированном автомате таких состояний может быть не одно а несколько. Ясно, что класс детерминированных автоматов Мили является подклассом класса недетерминированных автоматов.

Автоматы Мура. Автоматы Мура есть частным случаем автоматов Мили.

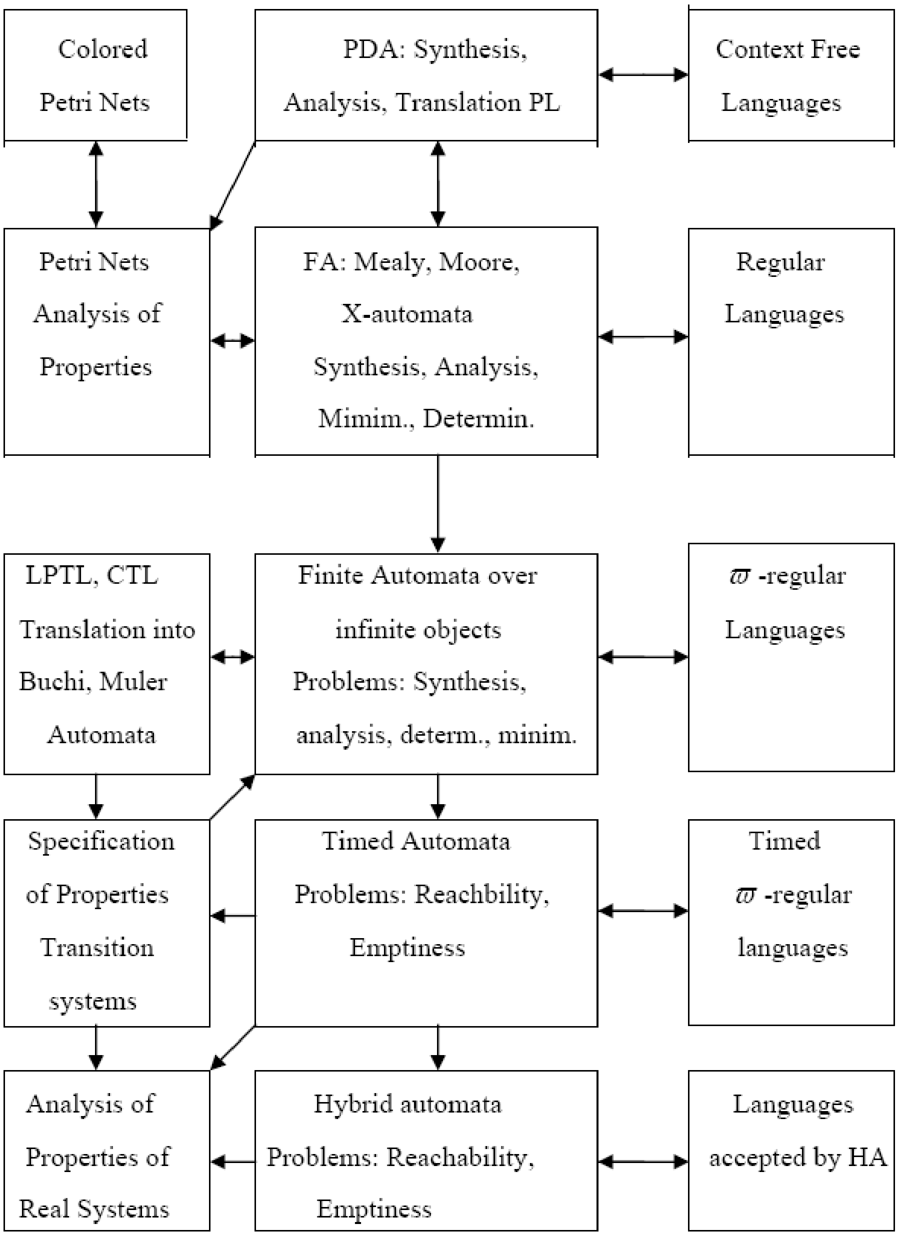
Определение 2. Автомат *(A, X, Y, f, g)* называется автоматом Мура, если его функция выходов *g(a,x)* выражается через функцию переходов *f(a,x)* при помощи уравнения *g(a,x)* = *h(f(a,x)),* где $h: A \rightarrow Y$. Функция *h* называется функцией отметок автомата, а ее значение *h(a)* на состоянии *a* называется отметкой этого состояния.

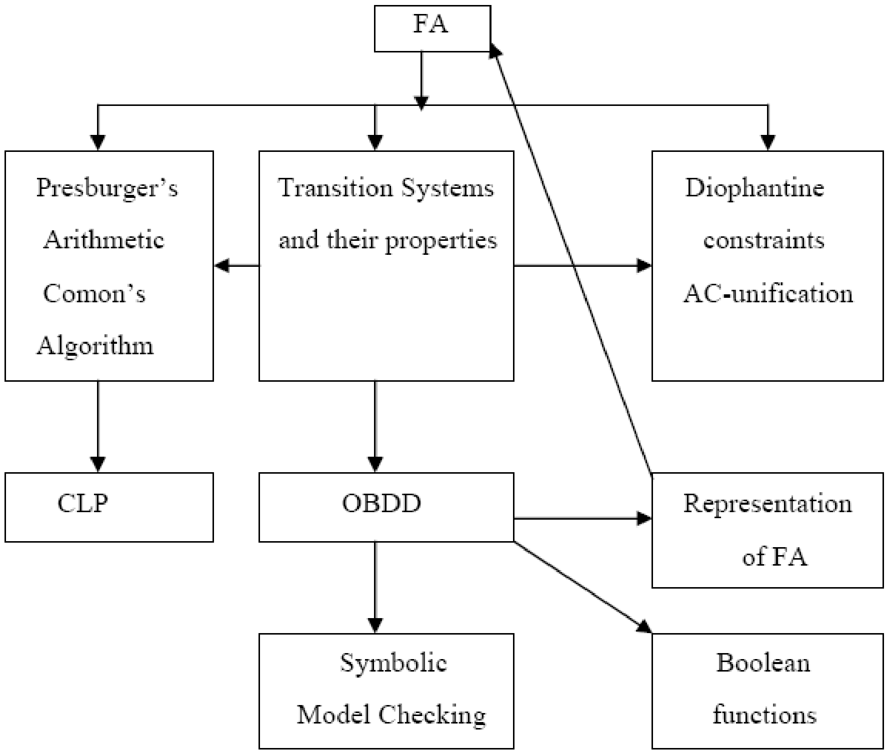
Несмотря на то, что автоматы Мура есть частным случаем автоматов Мили, в теории автоматов они являются объектом отдельного изучения, поскольку в ряде случаев их специфические свойства дают возможность строить более содержательную теорию, чем теория автоматов Мили.

X-автоматы. X-автоматом называется четверка *(A, X, f, F)*, а если *X*-автомат инициальный, то пятерка *(A, X, f,* a_0*, F)*, где $f\colon A \times X \to A$ - функция переходов, $F \subseteq A$ - некоторое подмножество состояний автомата, элементы которого называются заключительными состояниями, а $a_0 \in A$ – начальное состояние автомата.

3. Краткая онтология конечных автоматов

Ниже представлена краткая онтология теории автоматов, которая приведена в виде графа. Дуги этого графа связывают различные области, которые примыкают к той или иной теории автоматов или к ее приложениям.





4. Конечные автоматы над бесконечными словами

Пусть X – конечный алфавит. ϖ -языком называется произвольное подмножество X^{ϖ} - всех бесконечных слов в алфавите X. ϖ -автомат является конечным объектом для акцептации· ϖ -языков над некоторым алфавитом X. Существует несколько разновидностей ϖ -автоматов. Основными среди них есть автоматы Бюхи и Мюлера.

4.1. Автоматы Бюхи и автоматы Мюлера

Определение 3. Автоматом Бюхи A называется объект (A, X, f, A_0, F), где X – конечный входной алфавит, A - множество состояний автомата, $A_0 \subseteq A$ - множество начальных состояний, $f \subseteq A \times X \times A$ - множество дуг и $F \subseteq A$ - множество заключительных состояний. Объект (A, X, f, A_0) называется таблицею переходов.

Автомат стартует в одном из начальных состояний и если $(s, x, s') \in f$, то автомат может изменить сове состояние из s на s' путем чтения входного символа $x \in X$. Для слова $\sigma = x_1 x_2 x_3 \ldots$ в алфавите X будем говорить, что $r = s_0 x_1 s_1 x_2 s_2 x_3 \ldots$ является трассой автомата A над σ, которая соединяет $s_0 \in A_0$ и $(s_{i-1}, x_i, s_i) \in f$ для всех $i \geq 1$. Для трасы r, множество $inf(r)$ состоит из состояний $s \in A$ таких, что $s = s_i$ для бесконечного числа раз, $i \geq 0$. Трасса r автомата A над словом $\sigma \in X^{\varpi}$ называется акцептирующей трассой, тогда и только тогда, когда $inf(r) \cap F \neq \varnothing$. Другими словами, трасса r является акцептирующей тогда и только тогда, когда некоторое состояние из множества F повторяется бесконечно часто на трассе *r*. Язык *L(A)* называется акцептированным автоматом *A*, если он состоит из слов σ ϖ таких, что автомат *A* имеет трассу *r*, которая акцептирует σ .

ϖ -язык называется регулярным ϖ -языком, если он акцептируется некоторым автоматом Бюхи.

Пусть $L(A) = \{p \in X^{\varpi} \mid A$ акцептирует $p\}$ - ϖ -язык, который акцептируется некоторым автоматом *A*. Если $L = L(A)$ для некоторого автомата Бюхи *A*, то говорят, что язык *L* является Бюхи акцептируемым.

Обобщением автоматов Бюхи относительно множества заключительных состояний являются автоматы Мюлера.

Определение 4. Автоматом Мюлера над алфавитом *X* называется объект $A = (A, X, f, a_0, F)$, где *X* - конечный алфавит, *A* - конечное множество состояний, $a_0 \in A$ – начальное состояние, $f \subseteq A \times X \times A$ - отношение переходов, $F \subseteq B(A)$ - множество подмножеств множества заключительных состояний. Говорят, что трасса $\sigma = s_0 x_1 s_1 x_2 s_2 x_3 \ldots$ автомата является эффективной, если некоторые состояния этой трассы появляются бесконечное число раз и являются состояниями из множества *F*. Автомат A акцептирует слово $p \in X^{\varpi}$, если трасса, которая соответствует слову *p*, является эффективной. ϖ -язык $L \subseteq X^{\varpi}$ называется акцептованным автоматом Мюлера, если он состоит из всех ϖ -слов, которые акцептирует этот автомат.

4.2. Свойства регулярных ϖ -языков

Все результаты для ϖ -языков и ϖ -автоматов приведены в таблице 1. Приведенные в этой таблице результаты означают замкнутость соответствующих операций по отношению в данному классу автоматов

Таблица 1

Класс ϖ языков	Операции
МА = ВА = ДМА	Объединение, пересечение, дополнение
$\cup$	
ДВА	Объединение, пересечение

5. Временные автоматы

Пусть *B* - множество переменных часов, значения которых есть неотрицательными действительными числами *D+*. Множество временных ограничений *C(B)* определяется таким образом:

a) все элементы, принадлежащие к *C(B),* являются неравенствами и имеют вид $y \prec c$ и $c \prec y$, где $\prec$ означает < или $\leq$, а *c* – неотрицательное рациональное число;

b) если ϕ и ψ принадлежат к *C(B)*, то $\phi \wedge \psi \in C(B)$.

Заметим, что если *B* включает *k* часов, то каждое временное ограничение определяет некоторое выпуклое множество *k*-мерного евклидового пространства. Следовательно, если две точки удовлетворяют временному ограничению, то и все точки, которые лежат на отрезке, который соединяет эти точки, также удовлетворяют этому временному ограничению.

Определение 5. Временным автоматом (ВА) *A* называется шестерка (X, A, A_0, B, I, T), где *X* – конечный алфавит ВА, *A* – конечное множество состояний, $A_0 \subseteq A$ – множество начальных состояний, *B* –

множество часов, $I : A \rightarrow C(B)$ – отображение множества состояний во временные ограничения, которое называется инвариантами состояний, $T \subseteq S \times X \times C(B) \times 2^B \times A$ – множество переходов.

Каждая пятерка (a, x, ϕ, λ, a') соответствует переходу из состояния a в состояние a', который отмечен символом x $\in$X. Ограничение ϕ определяет момент времени, когда этот переход станет возможным, а значение часов из множества $\lambda \in$B сбрасываются в ноль при выполнении этого перехода**.**

5.1. Модель временного автомата

Моделью ВА *A* служит граф переходов $T(B) = (X, V, V_0, R)$ с бесконечным числом вершин. Каждая вершина из *V* представляет собой пару *(a, v)*, которая состоит из состояния $a \in A$ и значения часов $v : B \rightarrow D+$, которые они принимают в множестве неотрицательных рациональных чисел. Множество начальных вершин есть таким: $V_0 = \{(a,v) : a \in A_0 \wedge \forall y \in B[v(y) = 0]\}$.

Для определения переходов в *T(B)* необходимо ввести некоторую нотацию. Для $\lambda \subseteq B$ определим $v[\lambda = 0]$ как значения часов, в которых значения совпадают с *v* для часов из $B \setminus \lambda$ и приписывает значение 0 всем часам из λ.

Для $d \in D+$ определим *v + d* как значение часов, которые присваиваются всем часам из $y \in B$ значение *v(y) + d*, а значение часов *v - d* определяется аналогично.

Из определений следует, что ВА имеет два основных типа переходов:

- переход по задержке *d* соответствует определенному времени в том случае, когда автомат находится в некотором состоянии *a*. При этом пишут *(a, v){d} (a, v + d)*, где $d \in D+$, в том случае, если для всех $0 \le e \le d$ инвариант *I(a)* истинен для *v + e*;
- переход по действию соответствует выполнению перехода из множества *T*. В этом случае пишут, что *(a, v){x}(a', v')*, где $x \in X$, в том случае, если существует такой переход $(a, x, \phi, \lambda, a')$, что v удовлетворяет ϕ и $v' = v[\lambda = 0]$.

Отношение переходов *R* для *T(B)* получается путем объединения множества переходов по задержке с множеством переходов по действию. Запись *(a,v)R(a', v')* или *(a, v){x}(a', v')* означает, что существуют такие элементы *a''* и *v''*, что *(a, v){d} (a'', v''){x} (a', v')* , для некоторого $d \in D+$ и $x \in X$.

5.2. Временные языки

Пусть $X = \{x_1,\ldots,x_n\}$ алфавит и *R* – множество рациональных чисел. Временной последовательностью $\tau = \tau_1 \tau_2 \ldots$ называется бесконечная последовательность значений времени $\tau \in R$ такая, что $\tau_i > 0$ и удовлетворяет таким условиям:

(1) Монотонность: τ растет строго монотонно, т. е. $\tau_i > \tau_{i+1}$ для всех $i \ge 1$;

(2) Прогресс: для каждого $t \in R$ существует некоторое $i \ge 1$ такое, что $\tau_i > t$.

Временным словом в алфавите *X* называется пара (σ, τ), где $\sigma = \sigma_1 \sigma_2 \ldots$ является бесконечным словом над *X*, а τ есть временной последовательностью. Временным языком над алфавитом *X* называется множество временных слов над *X*.

Временное слово (σ, τ) называется входным словом автомата и в каждый момент времени *i* входной символ σ_i определяет момент времени τ_i.

Операции на временных языках такие как объединение, пересечение и дополнение определяются так же, как и для регулярных языков.

Для временного языка *L* в алфавите *X* оператор *Untime(L)* представляет собой проекцию временного слова на его первую компоненту, т. е . *Untime(L)* есть ϖ -языком, который включает все слова σ такие, что (σ, τ)$\in L$ для некоторой временной последовательности τ. Основные результаты для языков и операций над ними приведены в таблице 2.

Таблица 2

Класс временных языков	Операции
ТМА = ТВА	Объединение, пересечение
$\cup$	
ДТМА	Объединение, пересечение, дополнение
$\cup$	
ДТВА	Объединение, пересечение

6. Гибридные автоматы

Понятие гибридного автомата является обобщением понятия временного автомата. Гибридный автомат (ГА) представляет собой комбинацию дискретной динамики конечного автомата и непрерывной динамики динамической системы. ГА является удобной математической моделью многих реальных систем таких, как цифровые компьютерные системы, которые взаимодействуют с аналоговым окружением в реальном времени. В частности, при помощи ГА моделируются распределенные процессы со смешанным временем, протоколы управления и контроля в промышленных изделиях, движущиеся объекты, роботы и т. п.

Основными проблемами в теории ГА есть две проблемы:

- проблема достижимости и более общая
- **проблема пустоты** ϖ -языка.

Приведем основные определения и результаты в теории ГА. Пусть $D_{\geq 0}$ означает множество неотрицательных действительных чисел, т. е. $D_{\geq 0} = \{x \in D \mid x \geq 0\}$.

Прямоугольные области. Для натурального числа $n > 0$, $n \in N$ подмножество из D^n называется областью или регионом. Ограниченная и замкнутая область называется компактной. Область $R \subseteq D^n$ называется прямоугольником, если она представляет собой декартовое произведение интервалов (возможно и неограниченных), предельные точки которых являются рациональными числами. R_i означает проекцию *R* на *i*-тую координату, т. е. $R = R_1 \times R_2 \times \ldots \times R_n$. Множество всех прямоугольных областей в D^n обозначается R^n **.**

Определение 6. *n*-мерным прямоугольным автоматом (ПА) *A* называется такая девятка *A = (G, X, init, inv, flow, pre, post, jump, obs)*, где *G = (V, E)* – конечный ориентированный граф, *X* – конечный алфавит символов, для наблюдения, трех функций $init : V \to R^n$, $inv : V \to R^n$, $flow : V \to R^n$, которые отмечают вершины графа *G,* четырех функций pre: $E \to R^n$, *post* : $V \to R^n$, *jump* : $E \to 2^{\{1,2,\dots,n\}}$, *obs*: $E \to X$, которые помечают дуги графа *G*.

ПА может иметь так называемые ε -переходи (пустые переходы). *n*-мерный прямоугольный автомат с ε переходами отличаются от приведенного выше тем, что функция *obs* есть такой: *obs* : $E \to X^{\varepsilon}$, где X^{ε} = $X \cup \{\varepsilon\}$.

Функция *init* определяет множество начальных состояний ПА. Если дискретное состояние начинается в вершине $v \in V$, то непрерывное состояние должно начинаться в начальной области *init(v)*.

Функции *pre, post, jump* ограничивают поведение ПА во время выполнения переходов по дугах графа *G*. Дуга *e = (v, w)* может быть пройдена только когда дискретное состояние остается в вершине *v* и непрерывное состояние переходит в область *pre (v)*. Для каждого числа *i* множество вершин, в которые возможен переход (множество *jump set)*, *i*-ой координате непрерывного состояния недетерминированным образом присваивается новое значение из интервала *post (e).* Для каждого $i \notin$ *jump (e)*, *i*-тая координата непрерывного состояния не изменяется и остается в *post (e)*.

Функция наблюдения *obs* идентифицирует каждое ребро перехода помеченного символом наблюдения из *X* или из X^{ε} .

Функции *inv* и *flow* ограничивают поведение ПА во время переходов. До тех пор, пока дискретное состояние остается в вершине *v*, непрерывное состояние недетерминировано следует плавной траектории (C^{∞}) внутри инвариантной области *inv(v),* чье начальное время остается в текущей области *flow (v)*.

ПА с ε -переходами может двигаться по ε -дугах во время выполнения переходов.

Если в определении ПА заменить области произвольными линейными областями, то полученный автомат называется линейным гибридным автоматом. Прямоугольные автоматы составляют подкласс класса вех линейных гибридных автоматов, в которых все определяемые области прямоугольные.

Инициализация и ограниченный недетерминизм. ПА *A* называется инициальным, если для каждой дуги *e=(v, w)* из *A* и каждой координаты $i \in [1,2,\dots,n]$ и $flow(v)_i \neq flow(w)_i$ имеет место включение $i \in$ *jump(e)***.**

Отсюда следует, что каким бы образом не изменялась *i*-тая непрерывная координата инициального ПА, которая задана *flow* функцией, то ее значение недетерминированным образом реинициализируется в соответствии с *post* функцией.

ПА *A* называется ограничено детерминированным, если

(1) для каждой вершины $v \in V$ области *init(v)* и *flow(v)* ограничены и

(2) для каждой дуги $e \in E$ и каждой координаты $i \in [1,\dots,n]$ з того, що $i \in$ jump(e) вытекает, что интервал post(e) ограничен.

Заметим, что из условия ограниченности недетерминизма не следует условие конечности ветвлений в состоянии. Отсюда только следует, что время переходов по дугах в ограниченной области является ограниченным.

7. Приложения автоматов для верификации систем

Конечные автоматы с успехом используются для моделирования параллельных и взаимодействующих реактивных систем. При таком моделировании или множество состояний автомата, или символы его входного алфавита представляют состояния моделируемой системы. Основным преимуществом при таком использовании автоматов для верификации систем есть то, что как модель самой системы, так и ее спецификация представляются одинаково. В такому случае модель Крипке непосредственно соотносится с ϖ -автоматом (Б'юхи или М'юлера), все состояния которой являются заключительными, а множество возможных поведений системы задается ϖ -языком *L(A)*, который акцептируется соответствующим автоматом *A*. При этом существует алгоритм, который транслирует произвольную формулу темпоральной пропозициональной логики в ϖ -автомат Если *AP* – множество атомарных высказываний, то в частности, модель Крипке (S, R, S_0, f), где $f : S \to 2^{AP}$, можно преобразовать в автомат $A = (A \cup a_0, X, Q, a_0, A \cup a_0)$, у которого входной алфавит есть булеаном множества атомарных высказываний $X = 2^{AP}$. При этом для каждой пары состояний $a, a' \in A$ отношение Q включает тройку (a,x,a') тогда и только тогда, когда $(a,a') \in R$ и $x = f(a')$, причем $(a_0,x,a) \in Q$ в том и только в том случае, когда $a_0 \in S_0$ и $x = f(a)$, где *R* – отношение следования в модели Крипке.

8. Примеры некоторых свойств и их спецификаций

Рассмотрим некоторую гипотетическую реактивную систему. Свойства таких систем делят на два класса:

- **свойства безопасности (*safety properties*),** которые говорят о том, что нечто нежеланное не появится никогда**;**
- **свойства живучести (*liveness properties*),** которые говорят о том, что нечто хорошее в конце концов появится в системе.

Свойства безопасности выражаются, как правило, с помощью формул темпоральной логики вида $\Box \neg p$, где формула *p* характеризует не желаемое событие (состояние) в системе. Примером свойства безопасности может служить свойство взаимного исключения (*mutual exclussion*):$\Box (\neg p \vee \neg q)$, где формулы *p* и *q* характеризуют состояния, в которых система никогда не может находится одновременно. Примером свойства живучести может служить свойство справедливости (*fairness*), которое говорит о том, что когда к системе поступает запрос, то на него когда-нибудь будет получен ответ.

Расширим этот набор свойств еще некоторыми свойствами, которые принадлежат к классу свойств живучести. Эти свойства выглядят так:

- гарантия (guarantee) говорит о том, что некоторое событие появится хотя бы один раз, однако не гарантирует его повторения. Это свойство выражается ЛПТЛ-формулой $\Diamond p$;

- объязательство (obligation) говорит о том, что формула *p* должна выполнятся всегда или формула *q* выполняется в том же состоянии, что и формула *p*. Это свойство выражается ЛПТЛ-формулой $\Box p \vee \Diamond q$;

- реакция (response) говорит о том, что событие, которое описывается формулой *p*, появляется бесконечное число раз. Это свойство выражается ЛПТЛ-формулой $\Box \Diamond p$;

- настойчивость (persistense) говорит о том, что после некоторой задержки наступает стабилизация события·, которое описывается формулой *p*. Это свойство выражается ЛПТЛ-формулой $\Diamond \Box p$;

- реактивность (reactivity) представляет собой дизъюнкцию свойств реакции и настойчивости. Это свойство выражается ЛПТЛ-формулой $\Diamond \Box p \vee \Box \Diamond p$;

- безусловная справедливость (unconditional fairness) говорит о том, что событие, которое описывается формулой *q*, появляется бесконечное число раз независимо от условия *p*. Это свойство выражается ЛПТЛ-формулой $\Box \Diamond p$;

- слабая справедливость (weak fairness) говорит о том, что когда формула *p* все время истинная, то формула *q* должна быть истинной бесконечно часто. Это свойство выражается ЛПТЛ-формулой □ *p* → □ ⊔ *q*;

- сильная справедливость (strong fairness) говорит о том, что когда формула *p* истинная бесконечно часто, то формула *q* должна быть истинной тоже бесконечно часто. Это свойство выражается ЛПТЛ-формулой □ ⊔ *p* → □ ⊔ *q* .

Литература

1. Ахо А., Хопкрофт Дж., Ульман Дж. Построение и анализ вычислительных алгоритмов. -М. :Мир.-1979. - 535 с.
2. Alur R., Dill D.L. A theory of timed automata. - Theoretical Computer Science. -1994. -126. - PP. 183-235.
3. Thomas W. Automata on infinite objects. Handbook on theoretical computer science. - 1990. - PP. 135-191.
4. Годлевский А.Б., Кривой С. Л. Трансформационный синтез эффективных алгоритмов с учетом дополнительных спецификаций. Кибернетика, - 1986. - N 6. - С.34 - 43.
5. Глушков В.М. Абстрактная теория автоматов. - Успехи математических наук. -1961. - 16. - вып. 5. - С. 3-62.
6. Глушков В.М. Синтез цифровых автоматов. - М: Физматгиз. - 1962. - 476 с.
7. Глушков В.М., Цейтлин Г.Е., Ющенко Е.Л. Алгебра, языки, программирование. -Киев: Наукова думка. -1985. 327с.
8. 8. Perrin D. Finite automata. In Handbook of Theoretical Computer Science. vol. 2, -Elsevier. -1990. -PP. 1-58.
9. Henzinger T.A., Kopke P. W, Puri A., Varaiya P. What's Decidable About Hybrid Automata? In the Proceed. of the 27-th Annual ACM Symposium on Theory of Computing (STOC 1995). - 1995. - PP. 373-382.
10. 10. Comon H. Constraint solving on terms: Automata techniques (Preliminary lecture notes). - Intern. Summer School on Constraints in Computational Logics: Gif-sur-Yvette, France, September 5-8. -1999. - 22 p.
11. Капитонова Ю.В., Кривой С. Л., Летичевский А. А., Луцкий Г.М. Лекции по дискретной математике. БХВ: Санкт-Петербург, 2004, 624 с.
12. Трахтенброт Б. А., Барздинь Я. М. Конечные автоматы (Поведение и синтез). -М.: Наука. -1970. - 400 с.
13. Чень Ч., Ли Р. Математическая логика и автоматическое доказательство теорем. -М.: Мир. -1973. - 256 с.
14. Arnold A. Finite Transition Systems: Semantics of Communicating Systems. -Paris: Prentiuce Hall. -1994. - 177 p.
15. Ben-Ari M. Mathematical Logic for Computer Science. Springer Verlag London Limited. - 2001.-305 p.
16. Emerson E.A. Temporal and modal logics. Handbook of Theoretical Computer Science: Elsevier. - vol. B. -1990. - PP.995-1072.
17. Peterson G.L. Myths about the mutual exclusion problem. - Information Processing Letters, - 1981. - v.12.-N 3. - P.115-116
18. Wolper P. Temporal logic can be more expressive. - Information and Control. -v. 99. -1983. -P. 56 - 72.
19. Clarke E.M., Grumberg Jr. O., Peled D. Model Checking. - The MIT Press: Cambridge, Massachusetts, London, England. -2001. -356 p.

Информация об авторах

С. Л. Кривой, Л. Е. Матвеева – Институт кибернетики им. В. М. Глушкова НАН Украины, Киев, Украина, e-mail: krivoi@i.com.ua

Е. А. Лукьянова – Таврический национальный университет им. В.И. Вернадского, Симферополь, Украина,

О. Седлецкая – Ченстоховский политехнический институт, Ченстохов, Польша