УДК 519.713.1

А.Н. ЧЕБОТАРЕВ

КОМПОЗИЦИОННЫЙ ПОДХОД К ПРОЕКТИРОВАНИЮ РЕАКТИВНЫХ АЛГОРИТМОВ

Ключевые слова: *модуль алгоритма, композиционная спецификация, язык L, Σ-автомат, сверхслово, операции соединения модулей, выделение состояния.*

ВВЕДЕНИЕ

Рассматриваемый процесс проектирования реактивного алгоритма состоит в построении формальной спецификации требований к его поведению и автоматическом переходе от такой (как правило, декларативной) спецификации к императивному (процедурному) представлению алгоритма. Спецификация требований к функционированию алгоритма представляет собой множество утверждений, записанных на формальном языке, которые должны быть справедливы в каждый момент работы алгоритма. В используемом подходе к проектированию такая спецификация состоит из двух частей: спецификации управляющей части алгоритма и спецификации операционной части. Операционная часть алгоритма реализуется в виде достаточно регулярной структуры (программной или аппаратурной), поэтому задачи спецификации и реализации этой части существенно проще аналогичных задач для управляющей части, которым и посвящена настоящая работа. Поскольку проектирование управляющей части алгоритма осуществляется с учетом ограничений, определяемых его операционной частью и средой, для оптимизации управляющей части эти ограничения необходимо также специфицировать.

Алгоритмы функционирования небольших систем, имеющих достаточно простое управление, могут быть специфицированы одним набором требований, т.е. набором инвариантов, которые должны быть истинными в любой момент работы системы. Однако поведение сложных систем, в особенности таких, функционирование которых со временем может существенно изменяться, специфицировать одним набором требований несравнимо сложнее. С ростом сложности специфицируемого алгоритма увеличивается количество инвариантов и сложность формул, их выражающих, что приводит к нелинейному росту сложности спецификации в зависимости от сложности алгоритма. Очевидно, что вероятность появления ошибки в такой спецификации также значительно возрастает, кроме того, может недопустимо увеличиться время синтеза алгоритма, экспоненциально зависящее от размера спецификации. Поэтому спецификацию целесообразно разбить на несколько фрагментов, специфицирующих отдельные части алгоритма функционирования системы, называемые модулями и соединяемые в один алгоритм на уровне графов переходов автоматов, полученных путем синтеза управляющих частей модулей. Построенную таким образом спецификацию назовем композиционной. Композиционная спецификация состоит из спецификации модулей и спецификации связей между ними.

Композиционные методы являются одним из способов борьбы со сложностью решения задач проектирования. Это относится как к задачам верификации, так и к задачам синтеза, чему посвящено много публикаций. В задачах синтеза для поведенческого описания реактивных алгоритмов на начальных этапах про-

ектирования наибольшее распространение получили автоматные модели, такие как диаграммы состояний (Statecharts) [1–3], композиции реактивных модулей [4, 5] или расширенных автоматов (EFSM, CEFSM и др.) [6, 7]. Каждая такая модель представляет собой развитие традиционной модели конечного автомата (Finite State Machine), допускающее ее структурное иерархическое описание. Расширение понятия автомата в таких моделях, как реактивный модуль, EFSM и другие, связано с введением в модель типизированных переменных, определяющих состояния модели, описаний действий, изменяющих значения переменных, и предикатов, определенных на состояниях модели. В настоящей работе этому соответствует представление модуля и всего алгоритма в виде композиции управляющего и операционного автоматов. Такое разделение на два компонента и использование конечно-автоматной модели для представления информации о поведении операционного автомата позволяет оптимизировать управляющий автомат в процессе его синтеза.

Недостатком диаграмм состояний является невозможность верификации свойств такого поведенческого описания, поэтому в работах [8–10] предлагается дополнять описание утверждениями какой-либо темпоральной логики. Предпочтение здесь отдается логикам с темпоральными операторами прошлого времени. Однако основной недостаток перечисленных подходов состоит в необходимости неформализованной разработки процедурного (автоматного) представления спецификации алгоритма на верхнем уровне его проектирования.

Предлагаемый в настоящей работе подход к проектированию реактивного алгоритма существенно отличается от упомянутых подходов, а именно: уровнем исходной спецификации, представляющей собой логическую спецификацию требований к функционированию проектируемого алгоритма; использованием процедуры синтеза [11], дающей процедурное описание алгоритма, гарантированно удовлетворяющее всем требованиям спецификации; композиционным подходом как к построению логической спецификации требований к функционированию алгоритма, так и к его синтезу. В основе такого проектирования лежит синтез автоматов, соответствующих модулям, и соединение их графов переходов, определяемое спецификацией связей между модулями. Соединение автоматных моделей на уровне их графов переходов рассматривалось и в [12], где состояния отождествляются в некоторых простых структурах графов. В настоящей работе предложены развитые операции соединения модулей, определенные как на уровне их спецификации, так и на уровне графов переходов соответствующих автоматов, что позволяет получать любые структуры из соединяемых компонентов. Такой подход упрощает процесс написания спецификации, что снижает вероятность допущения в ней ошибок, а также существенно уменьшает время синтеза процедурного представления алгоритма.

СИНТАКСИС И СЕМАНТИКА ЯЗЫКА СПЕЦИФИКАЦИИ

Для спецификации модулей и связей между ними используется логический язык L [13]. Язык L является фрагментом логики предикатов первого порядка с одноместными предикатами и фиксированной областью интерпретации, в качестве которой выступает множество $\mathbf{Z}$ целых чисел. Спецификация в языке L имеет вид формулы $\forall t\, F(t)$. Формула $F(t)$ строится с помощью логических связок из атомарных формул (атомов) вида $p(t+k)$, где p — одноместный предикатный символ, t — переменная, принимающая значения из множества $\mathbf{Z}$, рассматриваемого как множество моментов дискретного времени, k — целое

число, называемое рангом атома. Разность между максимальным и минимальным значениями рангов атомов в формуле называется ее глубиной.

При определении автоматной семантики языков спецификации эти языки и автоматы рассматриваются как формализмы для задания множеств сверхслов (бесконечных слов) в алфавите $\Sigma(\Omega)$, где $\Omega = \{p_1, \ldots, p_m\}$ — набор предикатных символов спецификации. Символы алфавита $\Sigma(\Omega)$ представляют собой двоичные векторы длины m.

Определим необходимые понятия, связанные со сверхсловами и автоматами.

Пусть Σ — конечный алфавит, $\mathbf{Z}$ — множество целых чисел, $\mathbf{N}^+ = \{z \in \mathbf{Z} \mid z > 0\}$, $\mathbf{N}^- = \{z \in \mathbf{Z} \mid z \le 0\}$. Отображения $u: \mathbf{Z} \to \Sigma$, $l: \mathbf{N}^+ \to \Sigma$ и $g: \mathbf{N}^- \to \Sigma$ называются соответственно двусторонним сверхсловом (обозначается $\ldots u(-2)u(-1)u(0)u(1)u(2)\ldots$), сверхсловом (обозначается $l(1)l(2)\ldots$) и обратным сверхсловом (обозначается $\ldots g(-2)g(-1)g(0)$) в алфавите Σ. Отрезок $u(\tau)u(\tau+1)\ldots u(\tau+k)$ двустороннего сверхслова u обозначается $u(\tau, \tau+k)$. Бесконечные отрезки $u(-\infty, k)$ и $u(k+1, \infty)$ называются соответственно k-префиксом и k-суффиксом двустороннего сверхслова u. Для $n \in \mathbf{N}^+$ слово $g(1-n)\ldots g(0)$ называется n-суффиксом обратного сверхслова g.

Определение 1. Конечный неинициальный $X-Y$-автомат A — это четверка $\langle X, Y, Q, \chi_A \rangle$, где X, Y, Q — конечные множества соответственно входных символов, выходных символов и состояний, а $\chi_A: Q \times X \times Y \to 2^Q$ — функция переходов автомата.

$X-Y$-автомат A называется квазидетерминированным, если для любых $q \in Q, x \in X, y \in Y$ $|\chi_A(q, x, y)| \le 1$. Квазидетерминированные $X-Y$-автоматы удобно рассматривать как детерминированные частичные автоматы без выхода, с входным алфавитом $\Sigma = X \times Y$. Такой автомат $A = \langle \Sigma, Q, \delta_A \rangle$, где $\delta_A: Q \times \Sigma \to Q$ — частичная функция, будем называть Σ-автоматом.

Определение 2. Σ-автомат $A = \langle \Sigma, Q, \delta_A \rangle$ называется циклическим, если для каждого $q \in Q$ существуют такие $\sigma_1, \sigma_2 \in \Sigma$ и $q_1, q_2 \in Q$, что $q_1 = \delta_A(q, \sigma_1)$ и $q = \delta_A(q_2, \sigma_2)$.

В дальнейшем под автоматом будем понимать циклический Σ-автомат $A = \langle \Sigma(\Omega), Q, \delta_A \rangle$, где Ω — множество двоичных переменных, кодирующих символы его алфавита.

Определение 3. Сверхслово $l = \sigma_1\sigma_2\ldots$ в алфавите Σ допустимо в состоянии q автомата A, если существует такое сверхслово состояний $q_0q_1q_2\ldots$, где $q_0 = q$, что для любого $i = 0, 1, 2, \ldots$ $q_{i+1} = \delta_A(q_i, \sigma_{i+1})$. Множество всех сверхслов, допустимых в состоянии q, обозначим $W(q)$. Сверхслово l допустимо для автомата A, если оно допустимо хотя бы в одном из его состояний. Множество всех сверхслов, допустимых для автомата A, обозначим $W(A)$.

Определение 4. Обратное сверхслово $\ldots\sigma_{-1}\sigma_0$ в алфавите Σ представимо состоянием q автомата A, если существует такое обратное сверхслово состояний $\ldots q_{-2}q_{-1}q_0$, где $q_0 = q$, что для любого $i = -1, -2, \ldots$ $q_{i+1} = \delta_A(q_i, \sigma_{i+1})$. Множество всех обратных сверхслов, представимых состоянием q, обозначим $P(q)$.

Аналогично для произвольного $k \in \mathbf{N}^+$ определим множество $W^k(q_i)$ всех слов длины k, допустимых в состоянии q_i, и множество $P^k(q_i)$ всех слов длины k, представимых состоянием q_i.

Пусть $A = \langle \Sigma, Q, \delta_A \rangle$ и $q_1, q_2 \in Q$, $\sigma \in \Sigma$. Тройку $\langle q_1, \sigma, q_2 \rangle$, такую, что $\delta_A(q_1, \sigma) = q_2$, назовем переходом в автомате A из состояния q_1 в состояние q_2, а символ σ — отметкой этого перехода. Будем говорить, что входное слово $r = \sigma_1 \ldots \sigma_n$ переводит состояние q' в состояние q'', если существует такое слово состояний $q_1 q_2 \ldots q_{n+1}$, что для каждого $i = 1, \ldots, n$ в автомате A имеется переход $\langle q_i, \sigma_i, q_{i+1} \rangle$ и $q' = q_1$, $q'' = q_{n+1}$.

Определение 5. Σ-автомат A называется автоматом с конечной памятью, если существует такое натуральное k, что для любого входного слова длины k все состояния автомата A, в которых оно допустимо, переводятся этим словом в эквивалентные состояния. Минимальное такое k называется глубиной памяти автомата.

Рассмотрим теперь формулу языка L как способ задания множества сверхслов.

Каждой формуле $F = \forall t F(t)$ ставится в соответствие множество моделей для этой формулы, т.е. множество таких интерпретаций, на которых F истинна. Пусть $\Omega = \{p_1, \ldots, p_m\}$ — множество всех предикатных символов, встречающихся в формуле F (сигнатура формулы). Интерпретация формулы F — это упорядоченный набор определенных на $\mathbf{Z}$ одноместных предикатов $\pi_1, \ldots, \pi_m$, соответствующих предикатным символам из Ω. Интерпретацию $I = \langle \pi_1, \ldots, \pi_m \rangle$ можно представить в виде двустороннего сверхслова в алфавите $\Sigma(\Omega)$, а множество всех моделей для F — в виде множества $M(F)$ двусторонних сверхслов в этом алфавите. Поэтому можно говорить об истинностном значении замкнутой формулы, т.е. формулы вида $\forall t F(t)$ или $F(\tau)$, где $\tau \in \mathbf{Z}$, на двустороннем сверхслове. Обозначим $W(F)$ множество 0-суффиксов всех моделей из $M(F)$, а $P(F)$ — множество 0-префиксов этих моделей. Таким образом, с каждой формулой $F = \forall t F(t)$ ассоциируется множество сверхслов $W(F)$ и множество обратных сверхслов $P(F)$.

Формула $F(t)$ с единственной свободной переменной t называется 0-ограниченной, если для любого $\tau \in \mathbf{Z}$ значения формулы $F(\tau)$ на всех двусторонних сверхсловах, с одинаковыми τ-префиксами, совпадают. Будем такие формулы использовать для задания множеств обратных сверхслов, а именно, 0-ограниченная формула $F(t)$ задает множество $R(F(t))$ 0-префиксов всех таких двусторонних сверхслов, на которых истинна $F(0)$. Обозначим $R^k(F(t))$ множество k-суффиксов всех обратных сверхслов из $R(F(t))$.

Автоматную семантику языка L определяет следующая теорема.

Теорема 1 [13]. Для каждой непротиворечивой формулы $F = \forall t F(t)$ сигнатуры Ω существует в общем случае недетерминированный неинициальный циклический Σ-автомат $A = \langle \Sigma(\Omega), Q_A, \delta_A \rangle$ с конечной памятью, для которого $W(A) = W(F)$.

Такой автомат назовем автоматом, специфицируемым формулой F.

Предполагается, что символы сигнатуры Ω разбиты на два класса: входные и выходные, определяющие входной и выходной алфавиты соответствующего $X - Y$-автомата.

КОМПОЗИЦИЯ АВТОМАТОВ НА УРОВНЕ ИХ ГРАФОВ ПЕРЕХОДОВ

В качестве автоматной модели модуля рассматривается детерминированный Σ-автомат $A = \langle \Sigma, Q, \delta_A \rangle$. Значения функции переходов в состоянии q будем задавать в виде множества обобщенных переходов из этого состояния. Каждый такой переход соответствует множеству переходов между одними и теми же состояниями, при которых генерируется один и тот же выходной символ соответствующего $X - Y$-автомата. Отметка такого перехода состоит из двух

частей: входного условия, т.е. булевой функции от входных переменных, задающей множество символов входного алфавита $X-Y$-автомата, и символа выходного алфавита (набора значений выходных переменных). Входные условия переходов задаются формулами языка L, получающимися заменой переменных соответствующими атомами ранга 0. Таким образом, условию вида $f(x_1,\ldots,x_m)$ соответствует формула $f(x_1(t),\ldots,x_m(t))$. Будем говорить, что переход удовлетворяет условию $f(t)$, если формула языка L, соответствующая входному условию этого перехода, имплицирует формулу $f(t)$. Если символ выходного алфавита задавать конституэнтой единицы от выходных атомов нулевого ранга, то обе части отметки перехода могут быть заданы одной формулой языка L. При этом переход $\langle q_1, \sigma, q_2\rangle$ в квазидетерминированном $X-Y$-автомате описывается формулой $F(t-1)\&f(t)$, такой, что $R(F(t))\subseteq P(q_1)$, $R(F(t-1)\& f(t))\subseteq P(q_2)$, а $f(t)$ задает отметку перехода σ.

Уточним некоторые используемые в дальнейшем понятия.

Предыстория — это характеристика текущего момента работы системы, представляющая собой последовательность вход-выходных символов, полученных системой от начала ее функционирования (для неинициальных систем в бесконечном прошлом) до рассматриваемого момента. Для спецификации F множество всех возможных предысторий функционирования специфицируемой ею системы определяется множеством обратных сверхслов $P(F)$. Если максимальная глубина формул спецификации F в языке L равна k_1, то в качестве предысторий, однозначно характеризующих текущее состояние системы, можно рассматривать слова длины $k\geq k_1$, являющиеся k-суффиксами обратных сверхслов из $P(F)$. Множество всех таких слов обозначим $P^k(F)$.

Способ (режим) функционирования системы — это функция, которая каждой допустимой предыстории ставит в соответствие множество допустимых ее продолжений [14]. При композиционной спецификации системы возможные режимы ее работы определяются спецификациями используемых модулей. В синтезированной по такой спецификации системе множество различных режимов ее работы определяется разбиением множества состояний системы на классы, соответствующие множествам состояний соединяемых модулей. Переход от одного режима функционирования к другому связан с событием, представляющим собой изменение значения некоторого условия с ложного на истинное. На уровне графов автоматов, соответствующих модулям, такой переход описывается соединением графов, задаваемым с помощью двух основных операций:

— межмодульный переход (ММП);

— отождествление состояний (ОТС).

Спецификация каждого модуля состоит из спецификации управляющего автомата и спецификации локальной среды, характеризующей взаимодействие управляющего автомата специфицируемого модуля с операционным автоматом и внешней средой.

Спецификация соединений между модулями осуществляется в терминах спецификаций исходных модулей, поэтому результат построения специфицированного автомата не зависит от порядка выполнения соединений модулей.

СОЕДИНЕНИЕ МОДУЛЕЙ С ПОМОЩЬЮ МЕЖМОДУЛЬНОГО ПЕРЕХОДА

При реализации межмодульного перехода из модуля A в него добавляются переходы, для которых указываются модуль, куда осуществляется переход, и состояние в этом модуле. Каждый добавляемый обобщенный переход задается

двумя условиями, относящимися к различным модулям. Одно из них, называемое условием выхода из модуля, имеет вид $F(t-1)\& f(t)$, где $F(t-1)$ — внутренняя часть условия, такая, что $F(t)$ выделяет одно или несколько состояний в модуле A, а $f(t)$ — внешняя часть, определяющая отметки добавляемых обобщенных переходов из этих состояний. Другое условие вида $F(t)$, называемое условием входа в модуль, определяет состояние в модуле, куда осуществляется переход. Внутренняя часть условия выхода из модуля — это формула, все предикатные символы которой принадлежат сигнатуре соответствующего модуля, а ранги ее атомов не превышают -1. Внешняя часть — формула, построенная из атомов нулевого ранга, которая может содержать предикатные символы, не принадлежащие сигнатуре модуля A. Все предикатные символы в условии входа в модуль принадлежат сигнатуре этого модуля.

Уточним понятие выделения состояния условием $F(t)$.

Определение 6. Формула $F(t)$ строго выделяет во множестве Q состояние q, если $P(q) \subseteq R(F(t))$ и для любого состояния $q_1 \in Q$, не удовлетворяющего этому условию, $P(q_1) \cap R(F(t)) = \varnothing$.

Если $F(t-1)$ — внутренняя часть условия выхода из модуля, то формула $F(t)$ выделяет состояния в указанном смысле.

Для формулы $F(t)$, задающей условие входа в модуль, выделяемое ею состояние q удовлетворяет условию $P(q) \cap R(F(t)) \neq \varnothing$, которое назовем условием слабого выделения состояний.

Дадим формальное определение операции межмодульного перехода.

Пусть Σ-автоматы $A = \langle \Sigma_A, Q_A, \delta_A \rangle$ и $B = \langle \Sigma_B, Q_B, \delta_B \rangle$, где $\Sigma_A = \Sigma(\Omega_A)$ и $\Sigma_B = \Sigma(\Omega_B)$, соединяются с помощью межмодульного перехода из модуля A в модуль B, задаваемого условием выхода из A, равным $F_1(t-1)\& f(t)$, и условием входа в B, равным $F_2(t)$. Пусть условие $F_1(t)$ строго выделяет в модуле A состояние $q_1 \in Q_A$, а условие $F_2(t)$ слабо выделяет в модуле B состояние $q_2 \in Q_B$. Результат операции ММП представляет собой автомат $C = \langle \Sigma(\Omega_C), Q_C, \delta_C \rangle$. Сигнатура Ω_C равна $\Omega_A \cup \Omega_B \cup Z$, где Z — множество входных предикатных символов, содержащихся в $f(t)$ и не принадлежащих $\Omega_A \cup \Omega_B$. Множество Q_C равно $Q'_A \cup Q'_B$, где $Q'_A \cap Q'_B = \varnothing$ и Q'_A взаимно однозначно отображается в Q_A, а Q'_B — в Q_B. Состояние из Q_C, соответствующее состоянию $q \in Q_A \cup Q_B$, обозначим q'. Функция переходов δ_C для всех состояний, отличных от q'_1, совпадает с функцией переходов в соответствующих состояниях автомата A или B. В состоянии q'_1 функция переходов определяется следующим образом. К переходам, соответствующим переходам из состояния q_1, добавляются переходы из q'_1 в q'_2 с отметками, определяемыми формулой $f(t)$, а входные условия отметок всех переходов, соответствующих переходам из q_1, умножаются на отрицание дизъюнкции всех входных условий отметок добавленных переходов. Если условие $F_1(t)$ выделяет несколько состояний, то функция переходов в соответствующих состояниях автомата C совпадает с определенной выше функцией переходов в состоянии q'_1.

Добавляемый переход, входное условие которого зависит только от входных переменных модуля A, назовем внутренне обусловленным переходом, а переход, входное условие которого содержит переменные, не принадлежащие сигнатуре модуля A, — внешне обусловленным. Очевидно, что добавленный внутренне обусловленный переход из состояния q'_1 заменяет имевшийся переход из состояния q_1 с таким же входным условием. При добавлении внешне обусловленного перехода все переходы, соответствующие переходам из состоя-

ния q_1, сохраняются, а ортогональность их входных условий и входного условия добавленного перехода обеспечивается умножением этих условий на отрицание входного условия добавленного перехода. В случае добавления внутренне обусловленного перехода из инициального автомата A некоторые его состояния могут стать недостижимыми из начальных состояний в результате замены переходов между состояниями внутри автомата переходами в другой модуль. Соответствующие им состояния в автомате C будут удалены после выполнения всех операций соединения модулей композиционной спецификации. Если автомат A инициальный, то начальное состояние автомата C будет соответствовать его начальному состоянию.

СОЕДИНЕНИЕ МОДУЛЕЙ ПУТЕМ ОТОЖДЕСТВЛЕНИЯ СОСТОЯНИЙ

Эта операция состоит в отождествлении состояний, принадлежащих двум различным модулям. Ее спецификация заключается в спецификации для каждого модуля условия перехода, имеющего вид $F(t-1)\& f(t)$, где $F(t-1)$ — внутренняя часть условия, такая, что $F(t)$ строго выделяет отождествляемое состояние в соответствующем модуле, а $f(t)$ — внешняя часть, построенная из атомов нулевого ранга, образованных с помощью входных символов из объединения сигнатур соединяемых модулей. Формула $f(t)$ определяет множество переходов из отождествляемого состояния, удовлетворяющих этой формуле. Внешняя часть условия перехода для одного модуля является отрицанием внешней части условия перехода для другого модуля.

Пусть при соединении автоматов $A = \langle \Sigma(\Omega_A), Q_A, \delta_A \rangle$ и $B = \langle \Sigma(\Omega_B), Q_B, \delta_B \rangle$ соответствующие условия перехода имеют вид $F_1(t-1)\& f(t)$ и $F_2(t-1)\& \neg f(t)$. Условие $F_1(t)$ строго выделяет в модуле A отождествляемое состояние q_1, а условие $F_2(t)$ строго выделяет в модуле B отождествляемое состояние q_2. Перед тем как эти состояния будут отождествлены, в модуле A удаляются все переходы из состояния q_1, удовлетворяющие условию $\neg f(t)$, а в модуле B — все переходы из состояния q_2, удовлетворяющие условию $f(t)$. При этом автоматы A и B рассматриваются как Σ-автоматы с входным алфавитом $\Sigma(\Omega_A \cup \Omega_B)$. После отождествления состояний переходы из отождествленного состояния, удовлетворяющие условию $f(t)$, определяются автоматом A, а переходы, удовлетворяющие условию $\neg f(t)$, — автоматом B. Отсюда видно, что при отождествлении состояний обеспечивается ортогональность входных условий переходов из них, т.е. отождествление не приводит к появлению новых недетерминированных переходов. Заметим, что здесь речь идет о недетерминированности $X-Y$-автоматов.

Множество состояний автомата C, полученного путем соединения автоматов A и B, может содержать состояния, недостижимые из отождествленного состояния. Такие состояния будут удалены после выполнения всех операций соединения модулей композиционной спецификации. Если автомат A инициальный, то его начальные состояния будут начальными состояниями автомата C.

Автомат C, полученный с помощью операции ОТС, эквивалентен автомату, полученному путем выполнения двух групп операций ММП, одна из которых определяет переходы из модуля A в B, а другая — из модуля B в A. Внутренние части условий выхода в каждой из этих групп равны соответственно $F_1(t-1)$ и $F_2(t-1)$. Внешние части условий выхода первой группы соответствуют отметкам обобщенных переходов из состояния q_2, удовлетворяющим формуле $\neg f(t)$, а внешние части условий выхода второй группы — отметкам обобщенных переходов из q_1, удовлетворяющим формуле $f(t)$. Условия входа в модуль B выде-

ляют в нем состояния, в которые осуществляются обобщенные переходы из состояния q_2, соответствующие условиям выхода первой группы, а условия входа в модуль A — состояния, в которые осуществляются обобщенные переходы из q_1, соответствующие условиям выхода второй группы.

Операция ОТС может использоваться двояко: как операция перехода от модуля A к модулю B и как операция двустороннего соединения модулей, обеспечивающая переход от модуля A к модулю B и обратно. В первом случае отождествляемое состояние в модуле B не достижимо из себя с помощью путей, начинающихся переходами, удовлетворяющими условию $\neg f(t)$, или становится таковым вследствие перехода из модуля B к другому модулю. Во втором случае в модуле B существуют пути из отождествленного состояния в себя, начинающиеся переходами, удовлетворяющими условию $\neg f(t)$. Если при возврате в отождествленное состояние значение условия $f(t)$ становится истинным, то осуществляется переход к модулю A.

Возможно отождествление нескольких состояний одного модуля с одним и тем же состоянием такого же количества копий другого и отождествление одного состояния с несколькими состояниями другого модуля.

Состояния q_1 и q_2 одного и того же автомата называются совместимыми, если все переходы из них, имеющие одинаковые входные условия, различаются только этими состояниями, т.е. становятся одинаковыми, если q_1 и q_2 в них заменить одним и тем же состоянием. Отождествление совместимых состояний является эквивалентным преобразованием автомата. Отождествление одного состояния модуля A с несколькими состояниями модуля B допускается только в том случае, когда отождествляемые состояния модуля B совместимы.

Как отмечалось, операцию отождествления состояний q_1 и q_2 можно заменить несколькими операциями ММП, что позволяет при соединении модулей ограничиться только операцией ММП, однако это может увеличить количество состояний в объединенном автомате и не всегда удобно при спецификации соединяемых модулей.

ТРЕБОВАНИЯ К ВИДУ СОЕДИНЯЕМЫХ АВТОМАТОВ

Если в спецификациях связей между модулями фигурирует несколько условий строгого выделения состояний в одном и том же модуле, то этот модуль должен быть представлен в таком виде, чтобы для каждого из этих условий он имел одно или более состояний, строго выделяемых этим условием. Пусть $F_1(t),\ldots,F_k(t)$ — все различные условия, строго выделяющие состояния в модуле A. Предполагается, что эти условия попарно ортогональны. Тогда автомат $A=\langle\Sigma,Q,\delta_A\rangle$, соответствующий модулю A, должен удовлетворять требованию, чтобы для каждой $F_i(t)$ $(i=1,\ldots,k)$ формула $\forall_{q\in Q}(P(q)\cap R(F_i(t))\neq\varnothing\rightarrow$ $\rightarrow P(q)\subseteq R(F_i(t)))$ была истинной. Это требование, при необходимости, обеспечивается эквивалентным преобразованием автомата A путем расщепления его состояний. Аналогичное требование должно выполняться и в автомате C для множества состояний Q'_A.

Для автомата с конечной памятью введем понятие глубины разделимости состояний как такое наименьшее натуральное k, что для любых двух его состояний q_1 и q_2 $P^k(q_1)\cap P^k(q_2)=\varnothing$. Заметим, что глубина разделимости состояний отличается от глубины памяти автомата только в том случае, если он имеет эквивалентные состояния. Таким образом, для автомата с глубиной разделимости состояний, равной k, в качестве предысторий, однозначно характеризующих

текущее состояние системы, можно рассматривать слова длины k. Пусть глубина разделимости состояний автомата A равна k. Вследствие добавления переходов в некоторые состояния из Q'_A автомата C множества слов длины k, представимых состояниями из Q'_A, могут отличаться от аналогичных множеств для соответствующих состояний автомата A. Поэтому условие $\forall_{q\in Q'_A}(P(q)\cap R(F_i(t))\neq\varnothing\rightarrow$ $\rightarrow P(q)\subseteq R(F_i(t)))$ может быть ложным, т.е. для некоторых состояний из Q'_A условия их строгого выделения могут не выполняться. Для исключения такой ситуации преобразование множества состояний Q'_A осуществляется в рамках автомата, полученного погружением автомата A в контекст, определяемый связями модуля A с другими модулями. Контекст — это множество последовательностей переходов, ведущих в состояния рассматриваемого модуля. Максимальная длина таких последовательностей называется глубиной контекста. Она определяется максимальной глубиной условий, строго выделяющих состояния в этом модуле. Рассмотрим, как формируются части контекста глубины k, определяемые соответственно операциями ММП и ОТС.

Пусть для ММП из модуля A в модуль B условие входа выделяет в автомате B состояние q. Соответствующее условие выхода из модуля A, представленное в виде множества переходов, ведущих в состояние q, образует часть контекста, определяемую этим межмодульным переходом.

Пусть операция ОТС задана условиями $F_1(t-1)\&\, f(t)$ и $F_2(t-1)\&\,\neg f(t)$, где последнее условие определено в модуле A. Тогда формула $F_2(t)$ выделяет состояние в автомате A, а условие $F_1(t)$ определяет последовательность переходов, ведущую в это состояние и образующую соответствующий контекст.

В целях уменьшения количества состояний в преобразованном автомате условие строгого выделения состояний можно ослабить следующим образом.

Формула $F(t-1)\&\, f(t)$ (как для ММП, так и для ОТС) строго выделяет состояние q, если $P(q)\cap R(F(t))\neq\varnothing$ и существует такая $F_1(t)$, что $P(q)\setminus R(F(t))\subseteq$ $\subseteq R(F_1(t))$ и $F_1(t-1)\rightarrow\neg f(t)$. Другими словами, условие $P(q)\subseteq R(F(t))$ может не выполняться, если все переходы в состояние q, определяемые обратными сверхсловами из $P(q)\setminus R(F(t))$, гарантируют ложность формулы $f(t)$.

ПРИМЕР КОМПОЗИЦИОННОГО ПРОЕКТИРОВАНИЯ

Рассмотрим процесс спецификации и проектирования управляющей части ретранслятора, принимающего от передатчика потенциально бесконечную последовательность двоичных слов в параллельном коде и передающего их приемнику в последовательном коде. Прием и передача слов синхронизируются тактирующим сигналом, поступающим от передатчика. При нормальной работе устройства в каждом такте осуществляется прием очередного слова от передатчика и передача предыдущего слова приемнику. Правильность приема слов в ретрансляторе и приемнике контролируется. В случае неправильного приема слова в ретрансляторе генерируется сигнал RW и в следующем такте осуществляется повторный прием этого же слова. При неверном приеме слова в приемнике он генерирует сигнал FP, поступающий в ретранслятор через один такт после передачи слова, неверно принятого приемником. В течение такта, прошедшего между передачей слова и поступлением сигнала о его неверном приеме, может быть передано очередное слово. Поэтому после поступления сигнала FP ретранслятор повторно передает два последних переданных слова. Если слово, непосредственно следующее за неверно принятым приемником словом, не было передано, то сначала оно должно быть передано, после чего повторно передаются два последних переданных слова. Слово,

переданное непосредственно после неверно принятого приемником слова, в приемнике не контролируется, поэтому сигнал FP о его неверном приеме прийти не может. Очевидно, что ретранслятор должен содержать буфер для хранения не менее двух принятых от передатчика слов.

Формализуем описание управляющей части ретранслятора в виде композиционной спецификации в языке L. Для этого определим действия, выполняемые ретранслятором, и соответствующие им управляющие сигналы. Основные действия включают в себя прием очередного слова, повторный прием слова, передачу очередного слова и повторную передачу неверно принятого приемником слова. Для обозначения приема и повторного приема слова будем использовать соответственно символы x и y, а для передачи и повторной передачи — соответственно v и w. Сигнал FP запоминается на триггере (в операционной части), который устанавливается в 0 сигналом w. Сигнал, поступающий с этого триггера, обозначим TFP. Таким образом, сигнатура предикатных символов спецификации имеет вид $\Omega = \{\text{RW, TFP}, x, y, v, w\}$, из которых RW и TFP — входные, а x, y, v, w — выходные. Композиционная спецификация состоит из спецификаций основного модуля, выполняющего прием и передачу слов, если сбои в приемнике отсутствуют, модуля MFP, выполняющего действия, связанные с повторной передачей слов вследствие поступления сигнала TFP, и спецификации связей между этими модулями.

Спецификация основного модуля

Входные: RW.

Выходные: x, y, v.

$$x(t) \leftrightarrow RW(t). \quad y(t) \leftrightarrow \neg RW(t). \quad v(t) \leftrightarrow RW(t).$$

Заметим, что спецификация представляет собой конъюнкцию приведенных формул, предваренную квантором всеобщности $\forall t$. Этой спецификации соответствует автомат с одним состоянием, граф переходов которого изображен на рис. 1.

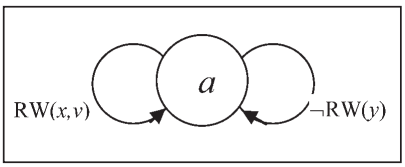

Рис. 1

Спецификация модуля MFP

Входные: RW, TFP.

Выходные: x, y, v, w.

$$y(t) \leftrightarrow \neg RW(t). \quad x(t) \leftrightarrow (\neg v(t-1)RW(t)TFP(t)).$$
$$w(t) \leftrightarrow (v(t-1)\neg w(t-1)TFP(t)).$$

$$v(t) \leftrightarrow (w(t-1) \vee \neg v(t-1)RW(t)TFP(t)).$$

$$w(t-1) \rightarrow \neg TFP(t). \quad (TFP(t-1)\neg w(t-1)) \rightarrow TFP(t).$$

Последние две формулы характеризуют поведение операционной части ретранслятора и обусловливают частичность синтезируемого автомата. Граф переходов автомата, соответствующего этой спецификации, приведен на рис. 2.

Рассмотрим сначала соединение модулей с помощью межмодульных переходов.

Необходимость перехода от функционирования основного модуля к модулю MFP возникает при наступлении события, соответствующего условию $TFP(t)$. Однако дальнейшее поведение модуля зависит от того, было ли передано очередное слово непосредственно перед поступлением сигнала TFP, а также от значения RW в момент поступления TFP, чему соответствует переход в различные со-

стояния модуля MFP. Условия выхода из основного модуля, характеризующие указанные ситуации, имеют следующий вид:

1) $RW(t-1)v(t-1)RW(t)TFP(t)w(t)$;
2) $RW(t-1)v(t-1)\neg RW(t)TFP(t)y(t)w(t)$;
3) $\neg RW(t-1)y(t-1)RW(t)TFP(t)x(t)v(t)$;
4) $\neg RW(t-1)y(t-1)\neg RW(t)TFP(t)y(t)\neg w(t)$.

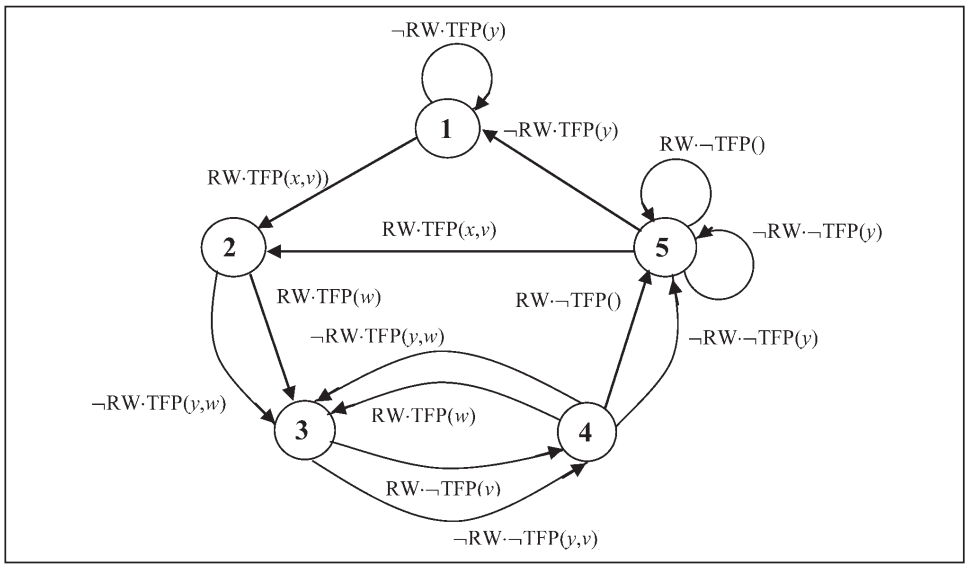

Рис. 2

Первому и второму условиям соответствует условие входа в модуль MFP, имеющее вид $TFP(t)w(t)$ и выделяющее состояние 3, третьему — условие входа $RW(t)TFP(t)x(t)v(t)\neg w(t)$, выделяющее состояние 2, а четвертому — условие $\neg RW(t)TFP(t)y(t)\neg w(t)$, выделяющее состояние 1. Для того чтобы внутренние части условий выхода строго выделяли соответствующие состояния в основном модуле, он преобразуется к виду, приведенному на рис. 3.

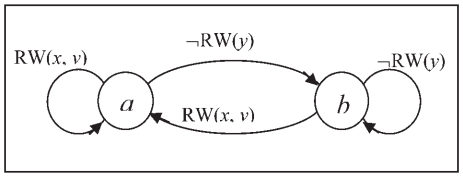

Рис. 3

В этом автомате условие $RW(t)v(t)$ строго выделяет состояние a, а условие $\neg RW(t)y(t)$ — состояние b.

Переход от модуля MFP к основному модулю специфицируется следующими условиями. Условия выхода из модуля MFP имеют вид

$$\neg TFP(t-1)v(t-1)RW(t)\neg TFP(t)x(t)v(t),$$

$$\neg TFP(t-1)v(t-1)\neg RW(t)\neg TFP(t)y(t).$$

Формула $\neg TFP(t)v(t)$, соответствующая внутренней части обоих условий, строго выделяет в модуле MFP состояние 4.

Первому из этих условий соответствует условие входа в основной модуль $RW(t)v(t)$, а второму — $\neg RW(t)y(t)$. Таким образом, соединение модулей осуществляется с помощью четырех операций ММП из основного модуля в модуль MFP и двух операций ММП из модуля MFP в основной модуль.

Имея графы переходов автоматов, соответствующих спецификациям модулей, и спецификации связей между ними, построим автомат, задаваемый приве-

денной композиционной спецификацией. Результатом первых двух операций является добавление двух переходов из состояния a основного модуля в состояние 3 модуля MFP соответственно с отметками $\mathrm{RW} \cdot \mathrm{TFP}(w)$ и $\neg\, \mathrm{RW} \cdot \mathrm{TFP}(y, w)$. При этом входные условия отметок всех имевшихся переходов из состояния a умножаются на $\neg$TFP. В результате следующих двух операций добавляются два перехода из состояния b основного модуля в состояния 2 и 1 модуля TFP соответственно с отметками $\mathrm{RW} \cdot \mathrm{TFP}(x, v)$ и $\neg\, \mathrm{RW} \cdot \mathrm{TFP}(y)$. Здесь также входные условия отметок всех имевшихся переходов из состояния b умножаются на $\neg$TFP.

В соответствии с операциями ММП из модуля MFP добавляются два перехода из состояния 4 этого модуля в состояния a и b основного модуля соответственно с отметками $\mathrm{RW} \cdot \neg\, \mathrm{TFP}(x, v)$ и $\neg\, \mathrm{RW} \cdot \neg\, \mathrm{TFP}(y)$. Входные условия отметок всех имевшихся переходов из состояния 4 умножаются на TFP. Таким образом, переходы из этого состояния в состояние 5 удаляются и оно становится недостижимым из остальных состояний модуля. Граф автомата, полученный в результате выполнения всех операций, соединяющих основной модуль и модуль MFP, приведен на рис. 4.

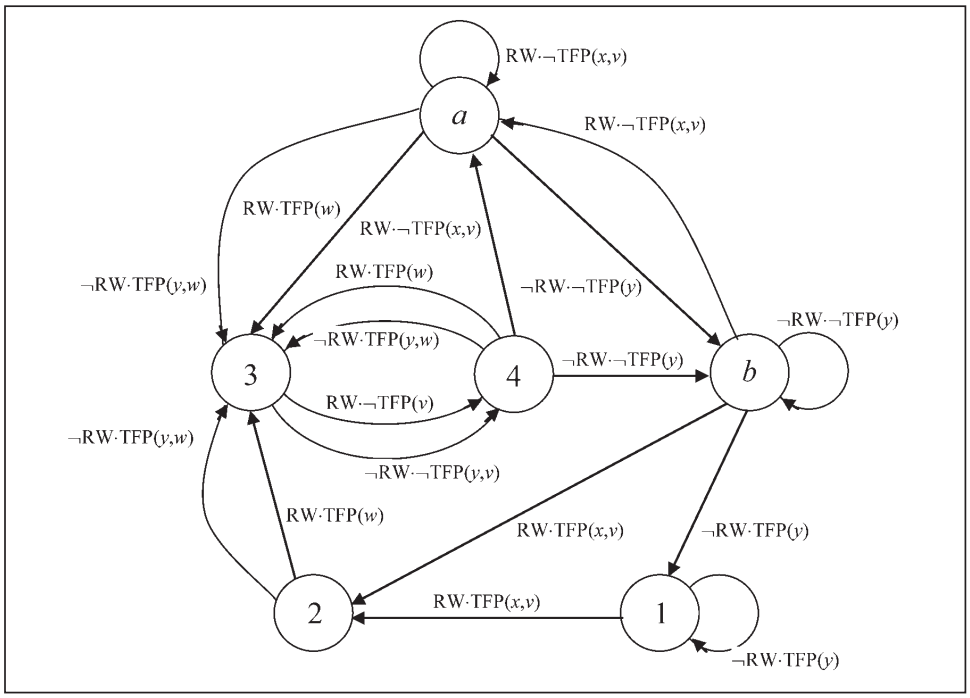

Рис. 4

Несложно заметить, что в этом автомате состояние 4 эквивалентно состоянию a. Состояния 2 и 1 частичные, и после подходящего доопределения они становятся эквивалентными соответственно состояниям 4 и b. Отметим, что произвольное доопределение этих состояний возможно, поскольку их частичность обусловлена взаимодействием с операционным автоматом. Отождествив эквивалентные состояния, получим автомат, изображенный на рис. 5.

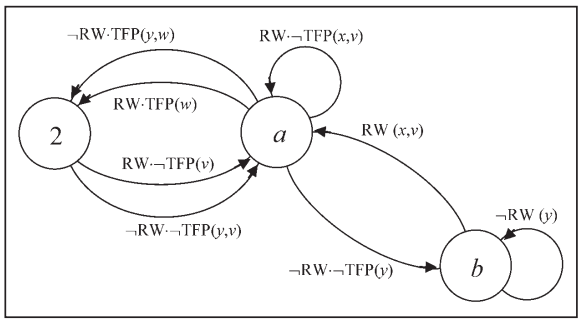

Рис. 5

Этот же результат можно получить более простым путем, используя две операции ОТС. Первая операция для основного модуля имеет условие перехода $RW(t-1)x(t-1)v(t-1)\neg TFP(t)$, а для модуля MFP — $\neg TFP(t-1)v(t-1)TFP(t)$, что соответствует отождествлению состояний a и 4, выделяемых внутренними частями соответствующих условий перехода. Во второй операции условие перехода для основного модуля имеет вид $\neg RW(t-1)\,y(t-1)\neg TFP(t)$, а для модуля MFP — $\neg RW(t-1)TFP(t-1)\,y(t-1)\neg w(t-1)TFP(t)$, что соответствует отождествлению состояний b и 1. Напомним, что соединяемые автоматы рассматриваются над объединенной сигнатурой, поэтому переходы в основном модуле рассматриваются как обобщенные переходы. Так, например, входное условие RW определяет два входных символа: RW·TFP и RW $\cdot\neg$TFP. При отождествлении состояний переходы из состояний основного модуля, удовлетворяющие формуле $TFP(t)$, удаляются. Для получения приведенного выше автомата необходимо отождествить состояния 2 и 4, что, вообще говоря, можно было сделать еще в автомате, соответствующем модулю MFP (рис. 2).

ЗАКЛЮЧЕНИЕ

В настоящей работе рассматривается композиционный метод проектирования реактивных алгоритмов в рамках автоматного подхода к проектированию [15]. В основе этого метода лежит понятие композиционной спецификации автомата и формальный переход от нее к его процедурному представлению. Композиционная спецификация связана с выделением различных режимов функционирования специфицируемого алгоритма, каждый из которых определяется спецификацией в языке L соответствующего автоматного модуля. Кроме того, такая спецификация определяет способы перехода от одного режима функционирования к другому. Каждый переход специфицируется парой формул языка L, относящихся к двум различным модулям и выделяющих в соответствующих автоматах состояния, используемые для их соединения. На автоматном (семантическом) уровне переходу от одного режима к другому соответствует операция соединения графов переходов автоматов.

Спецификация модуля состоит из двух частей: спецификации управляющего компонента модуля и локальной спецификации его операционной части и среды. Локальная спецификация — это спецификация тех свойств операционной части, которые не изменяются в течение функционирования модуля, хотя могут изменяться при переходе от одного модуля к другому. Использование локальной спецификации дает возможность осуществлять локальную оптимизацию частей графа переходов автомата, специфицируемого композиционной спецификацией. Такую оптимизацию целесообразно осуществлять до соединения модулей в один автомат, что упрощает решение задачи оптимизации всего специфицируемого автомата.

Результатом синтеза модуля является циклический автомат. При соединении модулей используется подавтомат этого автомата, выделяемый состояниями входа в модуль и состояниями выхода из него. Более того, этот автомат эквивалентно преобразуется к виду, удовлетворяющему требованиям, связанным с понятием строгого выделения состояний. Имеется достаточно простая формальная процедура такого преобразования, которая здесь не описана.

Спецификация отдельного модуля также может быть композиционной, что приводит к иерархической композиционной спецификации. Такое композиционное построение спецификации существенно упрощает ее написание и, следовательно, уменьшает возможность допущения в ней ошибок. Синтез управляющей части алгоритма сводится к синтезу более простых частей, что может на несколь-

ко порядков сократить суммарное время синтеза и значительно повысить качество получаемого решения за счет локальной оптимизации составляющих частей синтезируемого автомата.

СПИСОК ЛИТЕРАТУРЫ

1. Harel D. Statecharts. A upsilon isual formalism for complex systems // Sci. Comput. Program. — 1987. — **8**, N 3. — P. 231–274.
2. Harel D., Naamad A. The STATEMATE semantics of statecharts // ACM Trans. Software Eng. — 1996. — **5**. — P. 293–333.
3. Yi-Sheng Huang. Design of traffic light control systems using statecharts // Computer J. — 2006. — **49**, N 6. — P. 634–649.
4. Alur R., Henzinger T. Reacti upsilon e modules // Formal Methods in System Design. — 1999. — **15**, N 1. — P. 7–48.
5. Alur R., Grosu R. Modular refinement of hierarchic reacti upsilon e machines // Proc. 27th Ann. ACM Symp. Principles Programming Languages. — New York: ACM Press, 2000. — P. 390–402.
6. Petrenko A., Boroday S., Groz R. Confirming configurations in EFSM testing // IEEE Trans. Software Eng. — 2004. — **30**, N 1. — P. 29–42.
7. Byun Y., Sanders B.A. A pattern-based de upsilon elopment methodology for communication protocols // J. Inform. Sci. and Eng. — 2006. — **22.** — P. 315–335.
8. Sowmya A., Ramesh S. Extending statecharts with temporal logic // IEEE Trans. Software Eng. — 1998. — **24**, N 3. — P. 216–231.
9. Jemni L., Mosbahi O. Combining STATEMATE and FNLOG for the specification and the upsilon erification of complex real time systems // GESTS Intern. Trans. Comput. Sci. and Eng. — 2005. — **20**, N 1. — P. 65–76.
10. Drusinsky D. Semantics and runtime monitoring of TL Charts: Statecharts automata with temporal logic conditional transitions // Electron. Notes Theoret. Comput. Sci. — 2005. — **113**, N 3. — P. 3–21.
11. Тимофеев В.Г., Чеботарев А.Н. Усовершенствованный метод синтеза автомата по его спецификации в языке L // Кибернетика и системный анализ. — 2011. — № 3. — С. 3–14.
12. Byun Y., Sanders B.A, Chung K. A pattern language for communication protocols // Proc. 9th Conf. Pattern Languages Programs (PLoP'02). — Monticello (Ill.): Univ. of Illinois, 2002. — P. 1–32.
13. Чеботарев А.Н. Об одном подходе к функциональной спецификации автоматных систем. I // Кибернетика и системный анализ. — 1993. — № 3. — С. 31–42.
14. Чеботарев А.Н. О классе формул языка L*, специфицирующих автоматы с конечной памятью // Там же. — 2010. — № 1. — С. 3–9.
15. Чеботарев А.Н. Об одном способе спецификации реактивных алгоритмов в логическом языке первого порядка // Пробл. программирования. — 2000. — № 1, 2. — С. 273–279.

Поступила 16.04.2012