УДК 519.713.1

А.Н. ЧЕБОТАРЕВ, А.К. БОГАЧЕНКО

СИНТЕЗ АВТОМАТА, СПЕЦИФИЦИРОВАННОГО В ЯЗЫКЕ L*, ПУТЕМ ПЕРЕХОДА К СПЕЦИФИКАЦИИ В ЯЗЫКЕ L

A.N. Chebotarev, A.K. Bogachenko

SYNTHESIS OF AN FSM SPECIFIED IN THE LANGUAGE L* BY PASSING TO THE SPECIFICATION IN THE LANGUAGE L

A method for synthesizing an FSM specified in the logical language L is considered. The method is based on translating the specification into the language L and synthesizing an FSM from the specification in this language.*
Key words: reactive algorithms, language L, method of synthesis, FSM.

Розглядається метод синтезу скінченного автомата, специфікованого логічною мовою L. Метод базується на трансляції специфікації у мову L і синтезі автомата за специфікацією у цій мові.*
Ключові слова: реактивні алгоритми, мова L, метод синтезу, скінченний автомат.

Рассматривается метод синтеза конечного автомата, специфицированного в логическом языке L. Метод основан на трансляции спецификации в язык L и синтезе автомата по спецификации в этом языке.*
Ключевые слова: реактивные алгоритмы, язык L, метод синтеза, конечный автомат.

Введение. В основе доказательного проектированию реактивных алгоритмов лежит спецификация функциональных требований к алгоритму в языке логики первого порядка с одноместными предикатами и формальный переход от спецификации к процедурному представлению алгоритма. Одна из основных проблем, связанных со спецификацией свойств алгоритма, состоит в разработке подходящего языка спецификации. При этом приходится искать компромисс между выразительными возможностями языка и сложностью алгоритмов проектирования. Для разрешения этого противоречия используются два уровня языка: язык исходной спецификации L^* [1], обладающий достаточными для практических нужд выразительными возможностями и обеспечивающий удобство записи функциональных требований к алгоритму и язык L [2], обладающий сравнительно ограниченными выразительными возможностями, но эффективно обрабатываемый процедурами синтеза.

Автоматный подход к проектированию реактивного алгоритма характеризуется тем, что в качестве модели алгоритма используется сеть взаимодействующих автоматов. Таким образом, основные задачи проектирования связаны со спецификацией и синтезом конечных автоматов. Выразительные возможности языка L ограничены автоматами с конечной памятью [3], что позволило разработать для спецификаций в этом языке эффективные алгоритмы синтеза, проверки непротиворечивости спецификации и верификации темпоральных свойств, а также методы проектирования открытых систем, к которым относятся реактивные алгоритмы. Поэтому язык L^* используется (когда это необходимо) для начальной спецификации автоматов.

В основе рассматриваемого метода синтеза автомата по спецификации в языке L* лежит переход от этой спецификации к спецификации в языке L [4]. Такой переход дает спецификацию, не эквивалентную исходной, однако, как показано в [4], автомат, синтезированный по полученной спецификации, содержит подавтомат, совпадающий с автоматом, специфицированным в языке L*. Состояния, не принадлежащие этому подавтомату, называются фиктивными. Задача состоит в удалении из синтезированного автомата всех фиктивных состояний. В работе [5] предложен метод определения фиктивности состояний на основе проверки выполнимости формул языка L* в анализируемом состоянии. В настоящей работе на основе результатов, полученных в [6], предлагается фиктивность состояния определять путем проверки выполнимости в состоянии достаточно простой формулы языка L, что существенно уменьшает сложность процедуры определения фиктивности состояния.

Языки спецификации L и L^*.

Языки L^* и L построены на основе соответствующих фрагментов логики предикатов первого порядка с одноместными предикатами, определенными на множестве моментов дискретного времени, в качестве которого выступает множество Z целых чисел. Спецификация в обоих языках имеет вид формулы $\forall t F(t)$, где $F(t)$ – формула с одной свободной переменной t. В языке L формула $F(t)$ строится с помощью логических связок из атомов вида $p(t+k)$, где p – одноместный предикатный символ, t – переменная, принимающая значения из множества Z, а k – целочисленная константа (сдвиг во времени). Язык L^* отличается от языка L тем, что при построении формулы $F(t)$ наряду с атомарными формулами используются формулы вида

$$\exists t_i (t_i \le t + k_1) \,\&\, F_1(t_i) \,\&\, \forall t_j ((t_i + k_2 \le t_j \le t + k_3) \to F_2(t_j)),$$

где $k_1, k_2, k_3 \in Z$; $F_2(t_j)$ – формула языка L; $F_1(t_i)$ – формула языка L^*.

Такие формулы будем называть $\exists$-формулами. Для эквивалентного преобразования формул $F(t)$ такого вида часто используется равносильность [4]

$$F(t) \Leftrightarrow F(t-1) \,\&\, h(t) \vee g(t), \qquad (1)$$

где $F(t-1)$ обозначает формулу, полученную из $F(t)$ путем замены t на $t-1$, $h(t) = F_2(t+k_3)$, а $g(t) = F_1(t+k_1) \vee \ldots \vee F_1(t-k_2+k_3+1)$, если $k_3 < k_1 + k_2$, или $F_1(t+k_1) \,\&\, F_2(t+k_3) \,\&\ldots\&\, F_2(t+k_1+k_2)$, если $k_3 \ge k_1 + k_2$. Правую часть равносильности (1) назовем 1-разверткой $\exists$-формулы $F(t)$.

Для формулы $F(t)$ языка L^* следующим образом определим понятие *ранга* (обозначается $\#(F(t))$):

1) $(\#(p(t+k))=k$ (ранг атома);

2) если $F(t)=\exists t_i(t_i \le t+k_1)\,\&\,F_1(t_i)\,\&\,\forall t_j((t_i+k_2 \le t_j \le t+k_3)\to F_2(t_j))$, то $\#(F(t))=\max(r_1+k_1, r_2+k_3)$, где $r_1=\#(F_1(t_1))$, $r_2=\#(F_2(t_2))$;

3) если $F(t)$ построена из $F_i(t)$, $i=1,...,m$ с помощью логических связок, то $\#(F(t))=\max(\#(F_1(t)),\dots,\#(F_m(t)))$.

Объектом спецификации и синтеза является частичный, неинициальный, детерминированный автомат без выходов $A=<\Sigma, Q, \delta_A>$, где Σ – конечный входной алфавит, Q – конечное множество состояний, $\delta_A : Q\times\Sigma\to Q$ – частичная функция переходов. Такой автомат назовем Σ-автоматом. Автоматная семантика языков описывается в терминах циклических Σ-автоматов [4].

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

Поведение циклического Σ-автомата удобно описывать в терминах множеств сверхслов (ω-*слов*) в алфавите Σ, поэтому приведем связанные с ними определения.

Пусть Σ – конечный алфавит, Z – множество целых чисел, $N^+=\{z\in Z \mid z>0\}$ и $N^-=\{z\in Z \mid z\le 0\}$. Отображения $u: Z\to\Sigma, l: N^+\to\Sigma$ и $g: N^-\to\Sigma$ называются соответственно *двусторонним сверхсловом* (обозначается $\dots u(-2)u(-1)u(0)u(1)u(2)\dots$), *сверхсловом* (обозначается $l(1)l(2)\dots$) и *обратным сверхсловом* (обозначается $...g(-2)g(-1)g(0)$) в алфавите Σ. Отрезок $u(\tau)u(\tau+1)..u(\tau+k)$ двустороннего сверхслова u обозначается $u(\tau,\tau+k)$. Бесконечные отрезки $u(-\infty,k)$ и $u(k+1,\infty)$ будем называть соответственно *k-префиксом* и *k-суффиксом* двустороннего сверхслова u. Если значение k не существенно, то будем говорить об ω-*префиксе* и ω-*суффиксе*. Отрезок $l(1)..l(k)$ назовем *k-префиксом* сверхслова $l=l(1)l(2)....$

Сверхслово $l=\sigma_1,\sigma_2...$ в алфавите Σ *допустимо в состоянии* q Σ-автомата A, если существует такое сверхслово состояний $q_0q_1q_2\dots$, где $q_0=q$, что для любого $i=0,1,2,\dots$, $\delta_A(q_i,\sigma_{i+1})=q_{i+1}$. Множество всех сверхслов, допустимых в состоянии q, обозначим $W(q)$.

Обратное сверхслово $...\sigma_{-1}\sigma_0$ в алфавите Σ *представимо состоянием* q Σ-автомата A, если существует такое обратное сверхслово состояний $\dots q_{-2}q_{-1}q_0$, где $q_0=q$, что для любого $i=-1,-2,...$ $\delta_A(q_i,\sigma_{i+1})=q_{i+1}$. Таким образом, с каждым состоянием q циклического Σ-автомата ассоциируются два множества сверхслов: множество $W(q)$ всех сверхслов, допустимых в состоянии q, и множество $P(q)$ всех обратных сверхслов, представимых состоянием q.

Поведение автомата A с множеством состояний $Q_A = \{q_1, \ldots, q_n\}$ представляет собой семейство множеств сверхслов $S(A) = (W_1, \ldots, W_n)$, где $W_i = W(q_i)$, $(i = 1, \ldots, n)$.

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 Z одноместных предикатов $\pi_1, \ldots, \pi_m$, соответствующих предикатным символам из Ω. Интерпретацию $I = <\pi_1, \ldots, \pi_m>$ можно представить в виде двустороннего сверхслова в алфавите $\sum(\Omega)$, который представляет собой множество всех двоичных векторов длины m. Символы алфавита $\sum(\Omega)$ иногда удобно рассматривать как отображения вида $\sigma : \Omega \to \{0,1\}$. Пусть $\Omega_1 \subseteq \Omega$, *проекцией символа* $\sigma \in \sum(\Omega)$ на Ω_1 будем называть ограничение отображения σ на Ω_1. Понятие проекции символа на Ω_1 естественным образом распространяется на слова и сверхслова, так что проекция сверхслова в алфавите $\sum(\Omega)$ на Ω_1 есть сверхслово в алфавите $\sum(\Omega_1)$. В дальнейшем не будем различать интерпретации и соответствующие им двусторонние сверхслова, поэтому можно говорить об истинностном значении формулы F на двустороннем сверхслове u и значении формулы $F(t)$ в некоторой позиции τ двустороннего сверхслова u.

Каждой замкнутой формуле F ставится в соответствие множество $M(F)$ моделей для этой формулы, т. е. множество таких интерпретаций, на которых F истинна.

Обозначим $W(F)$ множество 0-суффиксов всех моделей из $M(F)$, а $P(F)$– множество 0-префиксов этих моделей.

Двустороннее сверхслово u' называется *k-сдвигом* ($k \in Z$) двустороннего сверхслова u, если для всех $i \in Z$ $u'(i) = u(i+k)$. Если u – модель для F, то и любой ее k-сдвиг принадлежит $M(F)$. Отсюда следует, что любой ω-суффикс модели для F принадлежит $W(F)$ и любой ее ω-префикс принадлежит $P(F)$.

Пусть $g \in P(F)$, обозначим $S(g)$ множество всех тех сверхслов, конкатенация каждого из которых с 0-префиксом g соответствует модели для F. Такие сверхслова назовем *допустимыми продолжениями* 0-префикса g, а k-префиксы сверхслов из $S(g)$ – *допустимыми k-продолжениями* g.

Формула F однозначно определяет конечную совокупность множеств сверхслов $S(F) = \{S_1, S_2, \ldots, S_m\}$ [4], такую, что для каждого $g \in P(F)$ $S(g) \in S(F)$ и, наоборот, для каждого $S_i \in S(F)$ существует $g \in P(F)$, для которого $S(g) = S_i$.

Так ω-префиксы g_1, g_2 моделей для F *эквивалентны*, если $S(g_1) = S(g_2)$. Пусть $P_1, P_2, \ldots, P_m$ – классы эквивалентности ω-префиксов из $P(F)$, соответствующие $S_1, S_2, ..., S_m$, тогда $\bigcup_{i=1}^{m} P_i S_i = M(F)$.

Формуле F поставим в соответствие *приведенный* автомат $A(F)$, состояния $q_1,...,q_m$ которого соответствуют классам $P_1, P_2, ..., P_m$, а функция переходов определяется следующим образом. Пусть g – произвольный 0-префикс из P_i и $\sigma_{i1},...,\sigma_{ik}$ – все допустимые 1-продолжения g, тогда $\delta(q_i, \sigma_{ij}) = q_{ij}$ $(j = 1, \ldots, k)$, где q_{ij} таково, что ω-префикс $g\sigma_{ij}$ принадлежит P_{ij}. Для всех символов алфавита $\sum(\Omega)$, отличных от $\sigma_{i1},...,\sigma_{ik}$, переход из q_i не определен. Очевидно, что для каждого q_i $(i = 1,...,m)$ существует, по крайней мере, один символ, для которого значение функции переходов определено и состояние q_j, из которого имеется переход в q_i. Следовательно, автомат $A(F)$ циклический.

Определение 1. Формула F специфицирует автомат A с поведением $S(A) = (W_1, W_2, ..., W_n)$ тогда и только тогда, когда существует такое разбиение $(P_1', P_2', \ldots, P_n')$ множества $P(F)$, что $\bigcup_{i=1}^{n} P_i' W_i = M(F)$.

Легко видеть, что любые два ω-префикса из P_i' $(i = 1,...,n)$ эквивалентны.

Доказательство. Допустим противное, т. е. $g_1 g_2 \in P_i'$ не эквивалентны. Это значит, что для одного из этих ω-префиксов, скажем g_2, существует допустимое продолжение l, не принадлежащее W_i (любое сверхслово из W_i – допустимое продолжение как для g_1, так и для g_2) и не принадлежащее $S(g_1)$. Тогда l принадлежит некоторому $W_j \neq W_i$, но поскольку $P_i' \cap P_j' = \varnothing$, то $g_2\ l$, не является моделью для F, а значит, и допустимым продолжением для g_2. Таким образом, приходим к противоречию.

Отсюда также следует, что каждое W_i $(i = 1,...,n)$ принадлежит $S(F)$. Таким образом, все автоматы, специфицируемые формулой F, строго эквивалентны автомату $A(F)$.

Пусть замкнутые формулы F_1 и F_2 имеют соответственно сигнатуры Ω_1 и Ω_2, а Ω – непустое подмножество множества $\Omega_1 \cap \Omega_2$. Формулы F_1 и F_2 назовем *эквивалентными относительно сигнатуры* Ω, если множества проекций всех моделей (двусторонних сверхслов) из $M(F_1)$ и $M(F_2)$ на Ω совпадают.

Переход от языка L* к языку L.

Преобразование исходной спецификации в спецификацию в языке L осуществляется в два этапа. На первом этапе спецификация $F = \forall t F(t)$ в языке L* преобразуется в такую эквивалентную относительно ее сигнатуры формулу

$F_1 = \forall t F_1(t)$ в языке L[*], которая специфицирует автомат с конечной памятью. Преобразование $F(t)$ в $F_1(t)$ осуществляется путем введения дополнительных предикатных символов для предикатов, удовлетворяющих $\exists$-подформулам исходной спецификации.

Пусть $\varphi_i(t)$ максимальная $\exists$-подформула формулы $F(t)$, т. е. подформула, не содержащаяся ни в какой другой $\exists$-подформуле. Каждая такая $\exists$-подформула заменяется атомом вида $z_i(t + r_i)$, где r_i – ранг заменяемой $\exists$-подформулы, а z_i – предикатный символ, отсутствующий в формуле $F(t)$, и в спецификацию добавляется эквивалентность $z_i(t) \leftrightarrow \varphi_i(t - r_i)$. Если $\exists$-формулы в правых частях эквивалентностей содержат отличные от них $\exists$-подформулы, то с ними поступают так как и с $\exists$-подформулами исходной формулы. Они заменяются соответствующими обозначениями новых предикатов и добавляются эквивалентности вида $z_j(t) \leftrightarrow \varphi_j(t)$. Так поступают до тех пор, пока ни одна из $\exists$-формул $\varphi_j(t)$ не будет содержать вхождений $\exists$-подформул, отличных от нее. В результате будет получена спецификация $F_1 = \forall t F_1(t)$. Как показано в [7] эта спецификация эквивалентна исходной спецификации относительно ее сигнатуры и специфицирует автомат с конечной памятью. На втором этапе в каждой эквивалентности $z_i(t) \leftrightarrow \varphi_i(t - r_i)$ $\exists$-формула $\varphi_i(t - r_i)$ заменяется правой частью равносильности (1), где обозначение формулы $\varphi_i(t - r_i - 1)$ заменяется на $z_i(t - 1)$. Таким образом, рассматриваемые эквивалентности приобретают вид $z_i(t) \leftrightarrow (z_i(t-1) \,\&\, h_i(t) \vee g_i(t))$, где $h_i(t)$ и $g_i(t)$ определяются 1-развертками соответствующих формул $\varphi_i(t)$. Это дает спецификацию $F_2 = \forall t f_z(t)$ в языке L. Второй этап представляет собой неэквивалентное преобразование, поскольку полученная формула F_2 языка L имеет дополнительные по сравнению с формулой F_1 модели, а синтезированный по ней автомат может иметь дополнительные состояния.

Фиктивные состояния.

Каждая модель из $M(F_2)$, не принадлежащая $M(F_1)$, имеет 0-префикс, не принадлежащий $P(F_1)$ [6]. Таким образом, любая модель для F_2, имеющая 0-префикс, принадлежащий $P(F_1)$, принадлежит $M(F_1)$. Отсюда следует, что $S(F_1) \subseteq S(F_2)$, т. е. любой автомат, специфицируемый формулой F_2, имеет подавтомат, строго эквивалентный автомату $A(F_1)$. Все состояния, не принадлежащие этому подавтомату, должны быть удалены. Это осуществляется путем удаления так называемых фиктивных состояний.

Определение 2. *Фиктивное состояние* автомата, специфицируемого формулой F_2 – это такое состояние, что все представимые им обратные сверхслова не являются 0-префиксами из $P(F_1)$.

Несложно убедиться в справедливости следующих утверждений.

Утверждение 1. Если 0-префикс g модели для F_2 принадлежит $P(F_1)$, то и любое обратное сверхслово gr, где r – допустимое *k*-продолжение g $(k \in N^+)$, также принадлежит $P(F_1)$.

Утверждение 2. Если 0-префикс g модели для F_2 не принадлежит $P(F_1)$, то и все его ω-префиксы не принадлежат $P(F_1)$.

Отсюда следует:

1) если состояние не фиктивно, то все достижимые из него состояния также не фиктивны;

2) если состояние фиктивно, то и все состояния, из которых оно достижимо, также фиктивны.

Это позволяет при анализе фиктивности состояний ограничиться только состояниями начальных сильно связных подавтоматов (НССП), т. е. таких сильно связных подавтоматов, состояния которых не достижимы из остальных состояний автомата. Очевидно, что либо все состояния НССП фиктивны, либо все они не фиктивны. Если автомат имеет фиктивные состояния, то имеется НССП, все состояния которого фиктивны. Все фиктивные состояния – это состояния, которые достижимы только из состояний таких НССП.

Пусть *u* – модель для формулы *F*, специфицирующей автомат *A* с поведением $(W_1, W_2, \dots, W_n)$, которому соответствует разбиение $(P_1', P_2', \dots, P_n')$ множества $P(F)$.

Утверждение 3. 0-префикс двустороннего сверхслова u представим состоянием автомата *A*.

Доказательство. Пусть u имеет 0-префикс из P_i'. Для каждого $k < 0$ *k*-префикс *u*(–∞, *k*) принадлежит некоторому классу из $\{P_1', P_2', ..., P_n'\}$. Таким образом, модель u определяет бесконечную последовательность классов из $\{P_1', P_2', ..., P_n'\}$, которым принадлежат *k*-префиксы двустороннего сверхслова u для всех $k \le 0$, а следовательно, и соответствующее обратное сверхслово состояний. Пусть для произвольного $k < 0$ $u(-\infty, k) \in P_l'$, а $u(-\infty, k+1) = u(-\infty, k)\sigma$ и принадлежит P_m'. Тогда согласно определению специфицируемого автомата *A* $\delta(q_1, \sigma) = q_m$. Отсюда следует, что 0-префикс *u*(–∞,0) представим состоянием q_i.

Пусть A_2 – автомат, специфицируемый формулой F_2. Тогда все ω-префиксы из $P(F_2)$ представимы состояниями автомата A_2 и для всякого состояния q, которым представим ω-префикс из $P(F_1)$, $W(q) \in S(F_1)$. Если состояния НССП автомата A_2 фиктивны, то удаление их сохраняет наличие в полученном автомате подавтомата, строго эквивалентного автомату $A(F_1)$.

Рассмотрим способ нахождения фиктивных состояний в автомате A_2.

С каждой формулой вида $z_i(t) \leftrightarrow (z_i(t-1) \,\&\, h_i(t) \vee g_i(t))$ в спецификации F_2 ассоциируется условие $c_i(t) = \neg g_i(t) h_i(t) z_i(t)$. Пусть в спецификации F_2 имеется k таких эквивалентностей. Как показано в [6] $M(F_1) \subseteq M(F_2)$, причем все модели для F_2, имеющие ω-префикс, в каждой позиции которого истинна одна из формул вида $\neg g_i(t) h_i(t) z_i(t)$, и только такие модели, не являются моделями для F_1.

Утверждение 4. Состояние q НССП фиктивно тогда и только тогда, когда все представимые им обратные сверхслова удовлетворяют одной и той же формуле $\forall t_1((t_1 \le t) \to c_i(t_1))$, т. е. во всех позициях таких обратных сверхслов истинна формула $c_i(t)$.

Достаточность непосредственно следует из определения фиктивности состояния. Для доказательства необходимости покажем, что если все обратные сверхслова, представимые состоянием q не принадлежат $P(F_1)$, то во всех позициях каждого такого сверхслова истинна одна и та же формула $c_i(t)$. Допустим противное, т. е. для каждого $c_i(t)$, $P(q)$ содержит обратное сверхслово, в некоторой позиции которого эта формула ложна. Поскольку q принадлежит сильно связному подавтомату, то существуют такие слова r_i $(i = 1, \ldots, k)$, переводящие состояние q в себя, что в некоторой позиции слова r_i ложна формула $c_i(t)$. Обратное сверхслово $(r_i \ldots r_k)^{-\omega}$, принадлежащее $P(q)$, не имеет ω-префикса, в каждой позиции которого истинна одна из формул $c_i(t)$. Таким образом, оно не принадлежит $P(F_2) \setminus P(F_1)$, т. е. принадлежит $P(F_1)$. Из этого следует, что состояние q не фиктивно. Полученное противоречие доказывает необходимость.

Проверка фиктивности состояний НССП основывается на следующем. Если формулы $c_i(t)$ $(i = 1, \ldots, k)$ содержат атомы только ранга 0, то состояния НССП фиктивны, если отметки всех дуг этого НССП имплицируют одну и ту же формулу $c_i(t)$. Если $c_i(t)$ имеет вид $c_{is}(t-s) \,\&\ldots\&\, c_{i0}(t)$, где формула $c_{ji}(t-j)$ построена из атомов ранга j, то она преобразуется в формулу $c_i'(t) = c_{is}(t) \,\&\ldots\&\, c_{i0}(t)$. Действительно, такая формула $c_i(t)$ может быть истинна во всех позициях обратного сверхслова только в том случае, если $c_i'(t)$ истинна во всех его позициях. Будем говорить, что формула $\forall t_1((t_1 \le t) \to c_i(t_1))$ выполняется на НССП, если отметки всех его дуг имплицируют формулу $c_i'(t)$. Таким образом, проверка фиктивности состояний НССП состоит в проверке выполнимости на нем каждой из формул $c_i(t)$ $(i = 1, \ldots, k)$. Фиктивные состояния удаляются из автомата. При удалении состояний какого-либо из НССП и всех состояний, не принадлежащих циклическому подавтомату, могут появиться новые НССП, для которых выполняется аналогичная проверка. В результате удаления всех фиктивных состояний будет получен автомат, строго эквивалентный автомату $A(F_1)$. Если будут удалены все состояния, то исходная спецификация F противоречива.

Выводы. Предложенный метод анализа фиктивности состояний и основанный на нем метод синтеза автомата, специфицированного в языке L^*, существенно проще метода, рассмотренного в [5], поскольку проверка фиктивности состояний основана на простых операциях пропозициональной логики.

1. *Чеботарев А.Н.* Расширение логического языка спецификации и проблема синтеза // Кибернетика и системный анализ. – 1996. – № 6. – С. 11–27.
2. *Чеботарев А.Н.* Об одном подходе к функциональной спецификации автоматных систем // Там же. – 1993. – № 3. – С. 31–42.
3. *Гилл А.* Введение в теорию конечных автоматов. – М.: Наука, 1966. – 227 с.
4. *Чеботарев А.Н.* Синтез процедурного представления автомата специфицированного в логическом языке L^*. Ч. I // Кибернетика и системный анализ. – 1997. – № 4. – С. 60–74.
5. *Чеботарев А.Н.* Синтез процедурного представления автомата специфицированного в логическом языке L^*. Ч. II // Там же. – 1997. – № 6. – С. 115–127.
6. *Чеботарев А.Н.* Преобразование спецификации автомата в языке L^* в автоматно эквивалентную спецификацию в языке L // Там же. – 2010. – № 4. – С. 60–69.
7. *Чеботарев А.Н.* О классе формул языка L^*, специфицирующих автоматы с конечной памятью // Там же. – 2010. – № 1. – С. 3–9.

Получено 01.09.2011