

На правах рукописи



Жигулин Максим Владимирович

**МЕТОДЫ СИНТЕЗА ПРОВЕРЯЮЩИХ ТЕСТОВ С
ГАРАНТИРОВАННОЙ ПОЛНОТОЙ ДЛЯ КОНТРОЛЯ ДИСКРЕТНЫХ
УПРАВЛЯЮЩИХ СИСТЕМ НА ОСНОВЕ ВРЕМЕННЫХ
АВТОМАТОВ**

05.13.01 – Системный анализ, управление и обработка информации
(в отраслях информатики, вычислительной техники и автоматизации)

Автореферат диссертации на соискание ученой степени
кандидата технических наук

Томск – 2012

Работа выполнена в Федеральном государственном бюджетном образовательном учреждении высшего профессионального образования «Национальный исследовательский Томский государственный университет» на кафедре информационных технологий в исследовании дискретных структур

Научный руководитель: доктор технических наук, профессор
Евтушенко Нина Владимировна

Официальные оппоненты: Матросова Анжела Юрьевна,
доктор технических наук, профессор,
ФГБОУ ВПО «Национальный
исследовательский Томский
государственный университет»,
зав. кафедрой программирования

Громаков Евгений Иванович,
кандидат технических наук, доцент,
ФГБОУ ВПО «Национальный
исследовательский Томский
политехнический университет»,
доцент кафедры интегрированных
компьютерных систем управления

Ведущая организация: ФГБУН Институт вычислительного
моделирования СО РАН,
г. Красноярск

Защита диссертации состоится 23 мая 2012 года в 10.30 на заседании диссертационного совета Д 212.267.12 при Томском государственном университете по адресу: 634050, г. Томск, пр. Ленина, 36, II уч. корпус, ауд. 212 б.

С диссертацией можно ознакомиться в Научной библиотеке Томского государственного университета.

Автореферат разослан 20 апреля 2012 г.

Ученый секретарь
диссертационного совета



Тарасенко
Петр Феликсович

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность проблемы. Тестирование является неотъемлемой частью жизненного цикла управляющих систем, цена ошибки в которых резко возрастает по мере нарастания сложности и важности решаемых задач. Существенный прогресс в области тестирования таких систем связывается с развитием методов синтеза тестов на основе формальных моделей, в частности, на основе конечно автоматных моделей. Поскольку классические автоматные модели явно не описывают временные аспекты поведения системы, такие как таймауты для ожидания подтверждения, допустимую задержку выходного отклика на поданное входное воздействие и т.п., в настоящее время активно исследуется вопрос о построении тестов с гарантированной полнотой на основе временных полуавтоматов и автоматов. Однако практически все известные методы синтеза тестов для временных автоматов доставляют конечные проверяющие тесты, полнота которых остается неизвестной, и, соответственно, синтез конечных проверяющих тестов с гарантированной полнотой для систем с явно выраженными временными аспектами требует дополнительных исследований. Достаточно выразительной моделью является конечный автомат с таймаутами, и, таким образом, разработка методов синтеза проверяющих тестов с гарантированной полнотой для таких временных автоматов является актуальной задачей.

Цель работы. Целью данной работы является разработка методов построения полных проверяющих тестов для контроля дискретных управляющих систем, поведение которых описано временными автоматами.

Методы исследования. Для реализации поставленной цели в работе используются средства и методы дискретной математики, в частности, методы теории автоматов. Оценка качества тестов, построенных предложенными алгоритмами, проводится с помощью компьютерных экспериментов.

Научная новизна работы состоит в следующем.

Предложен метод построения по временному автомату соответствующего конечного автомата и показано, каким образом можно использовать методы из классической теории автоматов при построении полных проверяющих тестов для временного автомата с использованием перехода к соответствующему классическому автомату. Однако также показано, что тесты, построенные по соответствующему конечному автомату относительно модели «черного ящика», оказываются сильно избыточными.

Исследовано отношение эквивалентности между детерминированными временными автоматами и установлены необходимые и достаточные условия

эквивалентности временных автоматов с таймаутами на основе соответствия между множествами их состояний.

Для понижения избыточности тестов относительно модели «черного ящика» методы построения полного проверяющего теста для конечного автомата адаптированы для построения тестов непосредственно по временному автомату.

Достоверность полученных результатов. Все положения, формулируемые в диссертации, доказываются с применением аппарата дискретной математики, в частности, элементов формальной логики и теории автоматов. Эффективность предложенных методов подтверждается посредством компьютерных экспериментов.

Практическая ценность. Разработанные алгоритмы и программное обеспечение могут быть использованы при построении проверяющих тестов для дискретных управляющих систем, поведение которых описано временными автоматами. В частности, результаты работы могут быть использованы в автоматизированных системах тестирования реализаций телекоммуникационных протоколов и другого программного обеспечения, для которых важны временные аспекты, и поведение которых можно описать соответствующим временным автоматом.

Реализация полученных результатов. Исследования, результаты которых изложены в диссертации, проводились в рамках следующих проектов:

НИР «Разработка математических и программных средств обеспечения надежного и безопасного доступа к электронным ресурсам коллективного пользования» (в рамках инновационного проекта ТГУ), 2006-2007 гг.

НИР «Проведение прикладных и проблемно-ориентированных поисковых исследований в области информационно-телекоммуникационных систем с участием научных организаций Франции» (шифр заявки «2009-04-1.4-00-02-003»), госконтракт №02.514.12.4002 от 09.06.2009г.

Основные положения, выносимые на защиту.

1. Метод построения по временному автомату соответствующего конечного автомата, который позволяет строить проверяющие тесты с гарантированной полнотой для временных автоматов на основе известных методов из классической теории автоматов.

2. Необходимые и достаточные условия эквивалентности двух временных автоматов на основе соответствия между множествами их состояний, которые позволяют адаптировать методы построения

проверяющих тестов относительно модели «черного ящика» из классической теории автоматов для построения проверяющего теста непосредственно по модели временного автомата, поскольку тесты, построенные по соответствующему классическому конечному автомату, обладают значительной избыточностью.

3. Метод построения полного проверяющего теста для временного автомата относительно модели «черного ящика» и разработанные на его основе алгоритмы построения полных проверяющих тестов относительно различных модификаций модели «черного ящика».

Апробация работы. Все теоретические и практические результаты, составившие основу диссертационной работы, по мере их получения обсуждались на семинаре кафедры информационных технологий в исследовании дискретных структур радиофизического факультета ТГУ. Кроме того, результаты работы докладывались на научных конференциях: VI (VII, VIII) Российской конференции с международным участием «Новые информационные технологии в исследовании дискретных структур», с. Шушенское, 2006 г. (Томск, 2008 г. и 2010 г.); Российской конференции «VI Всесибирский конгресс женщин-математиков», Красноярск, 2010 г.; Международной научно-практической конференции «Актуальные проблемы радиофизики», Томск, 2010 г.; Международной конференции по качеству программного обеспечения, QSIC'2011, Мадрид, Испания, 2011 г.

Структура и объем работы. Диссертация состоит из введения, 4 глав, заключения и списка используемой литературы. Диссертация содержит 13 рисунков. Объем диссертации составляет 121 страницу, в том числе: титульный лист – одна страница, оглавление – три страницы, основной текст – 99 страниц, библиография из 85 наименований – 10 страниц, приложение – 8 страниц.

Публикации. По результатам проведенных исследований опубликовано 9 статей в научных журналах и материалах конференций различного уровня. Работы [1-4] опубликованы в изданиях, входящих в список ВАК.

СОДЕРЖАНИЕ РАБОТЫ

Во **введении** дается общая характеристика работы, обосновывается актуальность исследований, определяется тематика и формулируется цель работы, кратко излагаются основные задачи и результаты, выносимые на защиту.

Первая глава посвящена основным понятиям и обозначениям, относящимся к конечным и временным автоматам, и, кроме того, содержит

обзор известных методов по синтезу проверяющих тестов для временных автоматов. Под *конечным автоматом (автоматом)* понимается пятерка $S = (S, I, O, T_S, s_0)$, где S – конечное непустое множество состояний с выделенным начальным состоянием s_0 ; I и O – конечные непустые входной и выходной алфавиты; $T_S \subseteq S \times I \times S \times O$ – *отношение переходов*; четверка «текущее состояние, входной символ, следующее состояние, выходной символ», называется *переходом* в автомате. В данной работе мы рассматриваем инициальные автоматы, то есть автоматы с фиксированным начальным состоянием; такие автоматы описывают поведение систем, обладающих сигналом сброса. Инициальный автомат описывает отображение входных последовательностей (слов) в выходные последовательности (слова). Если каждому входному слову соответствует единственное выходное слово, то автомат называется *детерминированным*: в противном случае автомат называется *недетерминированным*.

Далее, через \mathbb{N} обозначается множество натуральных чисел; через \mathbb{Z}_+ – множество неотрицательных целых чисел.

Временным автоматом называется семерка $S = (S, I, O, s_0, T_S, \Delta_S, \sigma_S)$, где S – конечное непустое множество состояний с выделенным начальным состоянием s_0 , I и O – конечные, непересекающиеся входной и выходной алфавиты, $T_S \subseteq S \times I \times O \times S$ – *отношение переходов*, $\Delta_S: S \rightarrow S \times (\mathbb{N} \cup \{\infty\})$ – *функция задержки*, определяющая время, по истечении которого состояние автомата может спонтанно измениться, если на автомат в текущем состоянии не поступил ни один входной символ, $\sigma_S: T_S \rightarrow \mathbb{N}$ определяет время задержки выходного символа после подачи входного воздействия на соответствующем переходе. Соответственно, поведение такого автомата зависит не только от входного символа, но и от момента времени, в который подается входной символ.

Временным входным символом называется пара $(i, t) \in I \times \mathbb{Z}_+$. Временной входной символ (i, t) означает, что входной символ i поступает на вход автомата в момент времени t после получения выходного символа на предыдущий входной символ. Для того чтобы определить выходную реакцию автомата на временной входной символ (i, t) в состоянии s , необходимо вычислить, в каком состоянии будет находиться автомат в t -й момент времени после перехода автомата в состояние s .

Временным выходным символом называется пара $(o, z) \in O \times \mathbb{Z}_+$. Временной выходной символ (o, z) означает, что выходной символ o появился на выходе автомата через z единиц времени после подачи соответствующего входного символа. Значение переменной z определяется

функцией σ_S для каждого перехода в автомате. Последовательность временных входных символов назовем *временной входной последовательностью*, и, соответственно, реакция автомата на временную входную последовательность, которая состоит из временных входных символов, называется *временной выходной последовательностью*.

Введем отображение $time_S: S \times Z_+ \rightarrow S$. По определению, $time_S(s, 0) = s$ для любого состояния s . Воспользовавшись функцией Δ_S , можно определить состояние $s' = time_S(s, t)$ как состояние, достижимое из состояния s по задержкам (таймаутам), пока сумма задержек из состояния s не превышает t , и становится больше t , если прибавить задержку в состоянии s' . Тогда во временном автомате существует временной переход $(s, (i, t), (o, z), s')$, если и только если во временном автомате существует переход $(time_S(s, t), i, (o, z), s')$, где значение z определяется функцией σ_S для данного перехода. Таким образом, множество $out_S(s, (i, t))$ выходных символов в состоянии $time_S(s, t)$ на входной символ i есть множество выходных реакций временного автомата в состоянии s на временной входной символ (i, t) . Множество $next_states_S(s, (i, t))$ есть множество состояний, в которые переводит входной символ i автомат S из состояния $time_S(s, t)$.

Обычным образом поведение временного автомата расширяется на временные входные и выходные последовательности. Пусть задана временная входная последовательность $\alpha = (i_1, t_1)(i_2, t_2) \dots (i_k, t_k)$ и $\beta = (o_1, z_1)(o_2, z_2) \dots (o_k, z_k)$ – временная выходная последовательность автомата. Четверка $(s, \alpha, \beta, s') \in T_S$, если и только если во временном автомате существует последовательность переходов $(s, (i_1, t_1), (o_1, z_1), s_1), (s_1, (i_2, t_2), (o_2, z_2), s_2), \dots, (s_{k-1}, (i_k, t_k), (o_k, z_k), s')$. В этом случае говорят, что временной автомат S в состоянии s *может выдать* временную выходную последовательность β на временную входную последовательность α и пара α/β называется *временной входо-выходной последовательностью* автомата.

Во временном автомате выделяют два вида состояний: *ir*-состояния, достижимые из начального состояния по некоторой временной входной последовательности, и *iur*-состояния, которые достижимы из некоторого *ir*-состояния только по задержкам. Временной автомат называется *связным*, если каждое состояние s' этого автомата является *ir*-состоянием или существуют *ir*-состояние s и такое t , что $time_S(s, t) = s'$. Мы рассматриваем только связные временные автоматы.

Для того чтобы определить правильность работы тестируемой системы, необходимо ввести отношение конформности. Такое понятие

обычно вводится на основе некоторого отношения между автоматом-спецификацией и проверяемым автоматом (автоматом-реализацией). В настоящей работе в качестве такого отношения выбрано отношение эквивалентности.

Пусть $S = (S, I, O, s_0, T_S, \Delta_S, \sigma_S)$ и $P = (P, I, O, p_0, T_P, \Delta_P, \sigma_P)$ – полностью определенные детерминированные временные автоматы. Будем говорить, что состояния s автомата S и p автомата P *эквивалентны* (обозначение: $s \cong p$), если для любой временной последовательности α справедливо $out_S(s, \alpha) = out_P(p, \alpha)$. В противном случае состояния называются *различимыми*.

Временной автомат S называется *приведенным*, если любые два различных состояния автомата различимы.

Временные автоматы S и P *эквивалентны* (обозначение: $S \cong P$), если эквивалентны их начальные состояния; в противном случае автоматы называются *различимыми*. Говорят, что временная входная последовательность α , для которой временные выходные последовательности автоматов S и P не совпадают, *различает* автоматы S и P .

Поскольку при определении отношения эквивалентности временные выходные последовательности автоматов на каждую временную входную последовательность должны совпадать, то определение временного автомата можно упростить, а именно, рассматривать временной автомат как шестерку $S = (S, I, O', s_0, T_S, \Delta_S)$ [7], в которой выходной алфавит O' содержит пары (o, z) , показывающие, в какой момент времени производится выходной символ o .

Как известно, синтез конечных проверяющих тестов с гарантированной полнотой возможен только в случае, когда число неисправностей в системе ограничено. Поэтому модель неисправности обычно содержит три элемента $\langle S, \cong, \mathfrak{F} \rangle$, где S – автомат-спецификация, которая описывает требуемое поведение проверяемой реализации системы, \cong – проверяемое отношение конформности и \mathfrak{F} – область неисправности, которая содержит автоматы, описывающие поведение каждой системы с неисправностями из заданного конечного множества. Под *полным проверяющим* тестом относительно заданной модели неисправности понимается конечное множество временных входных последовательностей такое, что для любого временного автомата из области неисправности, не эквивалентного автомату-спецификации, в тесте существует временная

входная последовательность, различающая этот автомат и автомат-спецификацию.

Самым простым способом задания модели неисправности является явное перечисление всех временных автоматов из области неисправности и нахождение для каждого неконформного автомата последовательности, отличающей этот временной автомат от автомата-спецификации. В общем случае такое описание области неисправности возможно только для небольшого числа неисправностей, и в случае, когда такое перечисление невозможно, в качестве модели неисправности часто рассматриваются модели «серого ящика» и «черного ящика». При использовании модели «серого ящика» предполагается, что частично известна структура проверяемой системы. При использовании модели «чёрного ящика» предполагается, что внутренняя структура проверяемой системы неизвестна. В модели «черного ящика» обычно предполагается известной верхняя оценка числа состояний проверяемого автомата, т.е. предполагается, что любая неисправность может увеличить число состояний только в некоторых пределах. В классической теории автоматов полные проверяющие тесты относительно модели «черного ящика» строятся на основе установления соответствия между состояниями и переходами автомата-спецификации и проверяемого автомата. Поскольку в этом случае проверяются все переходы в системе, то тесты, построенные относительно этой модели, являются достаточно качественными. Однако длина теста пропорциональна $|I|^{m-n+1}$, где $(m - n)$ равно разности числа состояний проверяемого автомата и автомата-спецификации и $|I|$ – мощность входного алфавита.

Если известно, что в проверяемой системе (проверяемом автомате) возможны только выходные ошибки, то в качестве полного проверяющего теста можно использовать обход графа переходов автомата-спецификации.

В главе 1 также приводится краткий обзор известных результатов по синтезу тестов для временных автоматов. Следует отметить, что все работы по синтезу тестов для временных автоматных моделей можно условно разбить на две группы. К первой группе относятся работы, в которых полный проверяющий тест строится на основе модели временного полуавтомата, введенного Алюром и Диллом, и отношения конформности *isco*. Однако все известные методы, доставляющие полные проверяющие тесты относительно такой модели неисправности, являются бесконечными. В публикации Спринджинтвелд и др. модель Алюра-Дилла несколько упрощается, и для упрощенной модели предлагается алгоритм построения конечного полного проверяющего теста относительно модели «черного ящика». Тест строится

W-методом в предположении, что поведение полуавтомата-спецификации и полуавтомата-реализации можно описать детерминированным конечным автоматом, и известна верхняя оценка числа временных областей для проверяемого полуавтомата. Однако как замечают авторы, получаемый в результате тест оказывается слишком избыточным, что затрудняет его использование на практике. Тем не менее, данная статья впервые показывает, что для временных автоматов/полуавтоматов можно построить конечный проверяющий тест с гарантированной полнотой даже в том случае, когда область определения временной переменной есть множество неотрицательных рациональных чисел, т.е. является бесконечной. Во второй группе представлены работы, в которых обсуждаются методы построения проверяющих тестов по модели временного автомата, как конечного автомата с таймаутами. В этих работах авторы рассматривают различные отношения конформности между такими автоматами, и, кроме того, устанавливают необходимые и достаточные условия различимости двух, возможно недетерминированных, временных автоматов на основе их пересечения. Было показано, что модель (расширенного) автомата с таймаутами является достаточно выразительной; с ее помощью можно описывать поведение различных временных систем, в частности, телекоммуникационных протоколов и web-приложений. Однако в большинстве работ авторы используют эвристические методы построения тестов, полнота которых остается неизвестной. Поэтому в данной работе мы продолжаем исследование автоматов с таймаутами с целью построения на основе этой модели проверяющих тестов с гарантированной полнотой.

Поскольку методы построения тестов с гарантированной полнотой хорошо развиты для классических конечных автоматов, то во **второй главе** диссертации (разделы 2.1 и 2.2) мы предлагаем метод преобразования временного автомата в соответствующий классический конечный автомат [10], который заключается в следующем. На первом шаге из исходного временного автомата удаляются все переходы по задержкам. Если в некотором состоянии значение таймаута равно t , то в строящемся классическом автомате добавляется $(t - 1)$ копия этого состояния. В каждой копии автомат имеет такое же поведение, как в оригинальном состоянии для всех входных символов. Вводятся дополнительный входной символ 1, обозначающий ожидание входного воздействия в течение одной единицы времени, и выходной символ n , обозначающий отсутствие выходного сигнала. Исходное состояние, его копии, и состояние, в которое автомат переходит по задержке из этого состояния, соединяются цепочкой переходов,

каждый из которых помечен входо-выходной парой $1/n$. Если таймаут в некотором состоянии равен «бесконечности», то в этом состоянии добавляется петля, помеченная входо-выходной парой $1/n$. Чтобы промоделировать задержки на выходе в классическом автомате каждый временной выходной символ заменяется абстрактным, представляющим пару «выходной символ, задержка выходного символа».

Показывается, что между временными входо-выходными последовательностями исходного временного автомата S и построенного классического автомата A_S существует взаимно однозначное соответствие.

Утверждение 1. Для любой временной входо-выходной последовательности $(i_1, t_1)/o_1 \dots (i_m, t_m)/o_m$ временного автомата S в соответствующем конечном автомате A_S существует входо-выходная последовательность $\underbrace{1/n \dots 1/n}_{t_1} i_1/o_1 \dots \underbrace{1/n \dots 1/n}_{t_m} i_m/o_m$.

Кроме того, для любой входо-выходной последовательности $\underbrace{1/n \dots 1/n}_{t_1} i_1/o_1 \dots \underbrace{1/n \dots 1/n}_{t_m} i_m/o_m$ автомата A_S в автомате S существует временная входо-выходная последовательность $(i_1, t_1)/o_1 \dots (i_m, t_m)/o_m$.

Теорема 1. Два временных полностью определенных автомата $S = (S, I, O, s_0, T_S, \Delta_S)$ и $P = (P, I, O, p_0, T_P, \Delta_P)$ эквивалентны, если и только если эквивалентны конечные автоматы A_S и A_P .

В разделе 2.3 мы рассматриваем тесты, построенные на основе обхода графа переходов соответствующего конечного автомата. Данный подход позволяет строить проверяющие тесты с гарантированной полнотой не только для управляющих систем, поведение которых описано детерминированным временным автоматом, но также и для систем, для которых соответствующий временной автомат является недетерминированным и/или частичным. В разделе 2.3 рассматривается проверяющий тест для протокола TFTP, построенный подобно тесту для протокола IRC [4], на основе обхода графа переходов соответствующего конечного автомата, который выявил ряд несоответствий в доступных реализациях этого протокола: класса **TFTPServer** из библиотеки `commons-net-2.0.0`, сервера **atftpd** для Linux. Кроме того, компьютерные эксперименты, проведенные со случайно сгенерированными временными автоматами, показали, что тесты, построенные непосредственно по временному автомату, короче тестов, построенных по соответствующему классическому автомату.

Как известно, тесты, построенные на основе обхода графа переходов, гарантированно обнаруживают только выходные ошибки. Для обнаружения

всех ошибок перехода системы под действием входного символа в «неправильное» состояние проверяющий тест должен строиться относительно модели «черного ящика».

В третьей главе обсуждаются методы построения проверяющих тестов для временных автоматов относительно модели «черного ящика». В разделе 2.2 показывается, что методы построения полного проверяющего теста для временного автомата относительно модели «черного ящика» по соответствующему классическому автомату обычно доставляют тесты с большой избыточностью [10]. Этот факт объясняется тем, что область неисправности для построенного конечного автомата содержит «лишние» автоматы, которые не соответствуют ни одному временному автомату из области неисправности. Поэтому в третьей главе мы адаптируем методы из классической теории автоматов для построения полных проверяющих тестов непосредственно по временному автомату.

Для этого первоначально устанавливаем конструктивные необходимые и достаточные условия эквивалентности двух детерминированных временных автоматов на основе соответствия между конечным множеством их состояний (раздел 3.2) [2, 7, 10].

Утверждение 2. Если $S = (S, s_0, I, O, T_S, \Delta_S)$ – приведенный автомат, то автоматы S и P эквивалентны, если и только если существует отображение $h: P \rightarrow S$, со следующими свойствами:

а) $h(p_0) = s_0$;

б) $\forall p \in P$ и $\forall i \in I$ выполняется $out_P(p, (i, 0)) = out_S(h(p), (i, 0))$ и $h(next_state_P(p, (i, 0))) = next_state_S(h(p), (i, 0))$;

в) $\forall p \in P_{ir}$ и любого момента времени t справедливо $h(time_P(p, t)) = time_S(h(p), t)$.

Следствие. Приведенные детерминированные полностью определенные временные автоматы S и P являются эквивалентными, если и только если существует взаимно однозначное отображение $h: S \rightarrow P$, такое, что $h(s_0) = p_0$ и для любого состояния $s \in S$, для каждого перехода $s - t \rightarrow s'$ и каждого перехода $s - i/o \rightarrow s'$ автомат P имеет переходы $h(s) - t \rightarrow h(s')$ и $h(s) - i/o \rightarrow h(s')$.

На основе установленного соответствия в разделе 3.3 предлагается метод синтеза полного проверяющего теста относительно модели «черного ящика» непосредственно по модели временного автомата [2, 3, 7, 9, 10]. В разделе 3.3.1 предполагается, что число состояний временного проверяемого автомата не превышает числа состояний автомата-спецификации, и конечная задержка в каждом состоянии не превышает некоторой величины B .

Предложенный метод синтеза полных проверяющих тестов основывается на W -методе построения тестов для конечных классических автоматов, однако имеет свои особенности.

Множество W временных входных последовательностей называется *множеством различимости* приведенного автомата S , если для любых двух различных состояний s_i и s_j существует временная входная последовательность $\gamma \in W$, такая что $outs_S(s_i, \gamma) \neq outs_S(s_j, \gamma)$.

Обозначим через S_{ir} множество ir -состояний, и через S_{iur} множество iur -состояний. Для каждого состояния $s \in S_{ir}$, построим временную входную последовательность α_s , которая переводит автомат S из s_0 в состояние s . Префикс замкнутое множество таких последовательностей есть *множество достижимости* V_{ir} для ir -состояний автомата-спецификации. Для достижения iur -состояний определим функцию $\upsilon: S_{iur} \rightarrow V_{ir} \times Z_+$, которая для каждого iur -состояния s' , определяет ir -состояние s , такое, что существует переход по задержке $time_S(s, \tau) = s'$, и возвращает пару $\{\alpha_s, \tau\}$, где $\alpha_s \in V$ переводит автомат S из s_0 в s , и τ – время задержки, через которое автомат S переходит из s в s' .

Соответственно, на первом шаге алгоритма с помощью множества W мы убеждаемся, что проверяемый автомат имеет такое же число состояний, что и автомат-спецификация. На втором шаге проверяются все переходы по входным символам. На третьем шаге проверяется соответствие задержки (таймаута) в каждом состоянии проверяемого автомата задержке в соответствующем состоянии автомата-спецификации. Для сокращения теста на последнем шаге из полученного множества удаляются последовательности, которые являются собственными начальными отрезками других последовательностей.

Алгоритм 1. Построение полного проверяющего теста для временных автоматов относительно модели неисправности $\langle S, \cong, \mathcal{I}_n(I, O, B) \rangle$.

Вход: Детерминированный полностью определенный приведенный временной автомат-спецификация $S = (S, s_0, I, O, T_S, \Delta_S)$ с n состояниями, B – максимальная конечная задержка в состояниях проверяемого автомата.

Выход: Полный проверяющий тест TS относительно модели неисправности $\langle S, \cong, \mathcal{I}_n(I, O, B) \rangle$.

Строим множества V_{ir} и W для автомата S .

Шаг 1. Строим множество $TS_1 = V_{ir} \cdot W$. Для каждого iur -состояния s' фиксируем пару $\upsilon(s') = \{\alpha_s, \tau\}$. Добавляем $\alpha_s \cdot W^\tau$ в множество TS_1 .

Шаг 2. Строим множество $TS_2 = V_{ir}(i, 0).W$ и для каждого iur -состояния s' , которое достижимо из зафиксированного состояния s по задержке τ , в множество TS_2 добавляем $\alpha_s(i, \tau).W$.

Шаг 3. $\forall s \in S_{ir}$ в множество TS_3 добавляем $\alpha_s.W^t$, $t = 1, \dots, T$, где $T = \Delta_S(s)_{\downarrow N}$ если $\Delta_S(s)_{\downarrow N} \neq \infty$ и $T = B$ в противном случае.

$\forall s' \in S_{iur}$, для которого $\upsilon(s') = \{\alpha_s, \tau\}$, в множество TS_3 добавляем $\alpha_s.W^{\tau+t}$, $t = 1, \dots, T$, где $T = \Delta_S(s')_{\downarrow N}$ если $\Delta_S(s')_{\downarrow N} \neq \infty$ и $T = B$ в противном случае.

Шаг 4. Из $TS = TS_1 \cup TS_2 \cup TS_3$ удаляем последовательности, которые являются собственными начальными отрезками других последовательностей.

Конец.

Теорема 2. Множество TS , построенное по алгоритму 1, есть полный проверяющий тест относительно модели неисправности $\langle S, \cong, \mathfrak{T}_n(I, O, B) \rangle$.

В разделе 3.3.3 предложенный метод построения теста распространяется на случай, когда число состояний в проверяемом автомате может превышать число состояний в автомате-спецификации. Для обеспечения покрытия всех переходов в проверяемом автомате строится специальное множество временных последовательностей, которые гарантируют обход всех переходов в проверяемом автомате, если последний отвечает так же как автомат-спецификация на все последовательности проверяющего теста, связанные с идентификацией состояний.

Утверждение 3. Пусть реакции автомата $P \in \mathfrak{T}_m(I, O, B)$ на последовательности множества $TS_1 \cup TS_2$ (алгоритм 1) совпадают с реакциями автомата-спецификации на эти последовательности и P' – подмножество состояний проверяемого автомата, для которых установлено соответствие по этим последовательностям. Пусть $s \in S_{ir}$ и $\alpha_s \in V$ переводит автомат S из s_0 в состояние s , и $\forall s' \in S_{iur}$ зафиксирована пара $\upsilon(s') = \{\alpha_s, \tau\}$. Тогда временные входные последовательности из множеств

$$\alpha_s.Traversal_{n,m} = \{(i_1, t_1). \dots .(i_l, t_l) : \sum_{j=1}^l \left\lfloor \frac{t_j}{B} \right\rfloor + l \leq m - n + 1\}$$

$$\alpha_s.Traversal_{n,m}^\tau = \{(i_1, \tau + t_1). \dots .(i_l, t_l) : \sum_{j=1}^l \left\lfloor \frac{t_j}{B} \right\rfloor + l \leq m - n + 1\}$$

покрывают все переходы из состояний множества P/P' автомата P .

Алгоритм 2. Построение полного проверяющего теста относительно модели неисправности $\langle S, \cong, \mathfrak{T}_m(I, O, B) \rangle$.

Вход: Детерминированный полностью определенный приведенный временной автомат-спецификация $S = (S, s_0, I, O, T_S, \Delta_S)$ с n состояниями, $m \geq n$ – максимальное число состояний в проверяемом автомате, B – максимальная конечная задержка в состоянии проверяемого автомата.

Выход: Полный проверяющий тест TS относительно модели неисправности $\langle S, \cong, \mathcal{I}_m(I, O, B) \rangle$.

Строим множества $V_{ir}, W, Traversal_{n,m}$ для автомата S .

Шаг 1. Строим множество $TS_1 = V_{ir} \cdot W$. Для каждого iur -состояния s' фиксируем пару $\upsilon(s') = \{\alpha_s, \tau\}$. Добавляем последовательности $\alpha_s \cdot W^c$ в множество TS_1 .

Шаг 3. Строим множество $Traversal$, равное объединению множеств $\alpha_s \cdot Traversal_{n,m}$, $\alpha_s \in V_{ir}$, и для каждого iur -состояния s' с зафиксированной парой $\upsilon(s') = \{\alpha_s, \tau\}$ в $Traversal$ добавляем множество $\alpha_s \cdot Traversal_{n,m}^c$. В множество TS_2 для каждой последовательности $\beta \in Traversal$ включаем все последовательности $\beta \cdot W$ и, кроме того, для каждой последовательности $\beta \cdot (i_k, t_k) \in Traversal$ в TS_2 добавляются все последовательности $\beta \cdot W^{t_k}$.

Шаг 4. Из $TS = TS_1 \cup TS_2$ удаляем последовательности, которые являются начальными отрезками других последовательностей.

Конец.

Теорема 3. Множество TS , построенное по алгоритму 2, есть полный проверяющий тест относительно модели неисправности $\langle S, \cong, \mathcal{I}_m(I, O, B) \rangle$.

В разделе 3.4 обсуждаются оценки максимальной длины проверяющих тестов, построенных предложенным методом, и показывается, что порядок верхних оценок длины построенных тестов совпадает с таковым для классических конечных автоматов. При одинаковом числе состояний проверяемого автомата и автомата-спецификации верхняя оценка длины полного проверяющего теста полиномиально зависит от размерности автомата-спецификации и максимальной величины конечного таймаута в состоянии проверяемого автомата. Если проверяемый автомат может иметь больше состояний, чем автомат-спецификация, то верхняя оценка длины полного проверяющего теста экспоненциально зависит от разности числа состояний в проверяемом автомате и автомате-спецификации. Проведенные эксперименты показывают, что порядок длины теста относительно модели «черного ящика» для случая, когда неисправности не увеличивают число состояний в тестируемой реализации, совпадает с порядком проверяющих тестов для классических конечных автоматов [8]. Более того, длина таких проверяющих тестов близка к длине входо-выходной последовательности,

которую нужно пронаблюдать при пассивном тестировании протокольной реализации [1, 2, 5, 6] для того, чтобы гарантировать полноту проверки относительно модели «серого ящика».

В **четвертой главе** рассматриваются модификации модели неисправности «черный ящик» [3, 9], и предлагается соответствующая адаптация алгоритмов, описанных в главе 3, при построении полных проверяющих тестов относительно таких моделей неисправности.

В первом случае (раздел 4.1) предполагается, что в проверяемом автомате в каждом из состояний возможны только задержки, которые есть в автомате-спецификации. Пусть задан автомат-спецификация S с n состояниями и множеством конечных задержек $\Pi = \{b_1, \dots, b_z\}$, $z \leq n$, т.е. $\Delta_S: S \rightarrow S \times (\Pi \cup \{\infty\})$. Мы предполагаем, что проверяемый автомат есть некоторый полностью определенный детерминированный временной автомат P из множества $\mathcal{T}_n(I, O, \Pi)$. т.е. автомат P имеет не более n состояний, входной и выходной алфавиты I и O , и конечная задержка в состоянии есть некоторый элемент множества Π , т.е. $\Delta_P: P \rightarrow P \times (\Pi \cup \{\infty\})$. Алгоритм построения теста для данного случая аналогичен алгоритму 1, только на шаге 3 значение $t \in \Pi$ и если $\Delta_S(s)_{\downarrow N} \neq \infty$, то $t \leq \Delta_S(s)_{\downarrow N}$.

Теорема 4. Множество TS , построенное по алгоритму 1 с указанным выше изменением, есть полный проверяющий тест относительно модели неисправности $\langle S, \cong, \mathcal{T}_n(I, O, \Pi) \rangle$.

Во втором случае (раздел 4.2) предполагается, что конечная задержка в каждом из состояний проверяемого автомата не меньше некоторой известной величины. Пусть задан приведенный автомат-спецификация S с n состояниями. Мы предполагаем, что проверяемый автомат есть некоторый временной автомат P , который имеет не более n состояний, входной и выходной алфавиты I и O , и конечная задержка в каждом состоянии автомата P не меньше D и не больше B . Обозначим такую область неисправности как $\mathcal{T}_n(I, O, [D, B])$. Алгоритм построения теста для данного случая аналогичен алгоритму 1, только на шаге 3 значение $t \in \{D, \dots, \Delta_S(s)_{\downarrow N}\}$, если $\Delta_S(s)_{\downarrow N} \neq \infty$ и $t \in \{D, \dots, B\}$ в противном случае.

Теорема 5. Множество TS , построенное по алгоритму 1 с указанным выше изменением, есть полный проверяющий тест относительно модели неисправности $\langle S, \cong, \mathcal{T}_n(I, O, [D, B]) \rangle$.

В разделе 4.3 показывается, что полный тест для проверки эквивалентности проверяемого автомата автомату-спецификации является достаточным и для проверки некоторых несоответствий по времени

обработки входных воздействий проверяемым автоматом. В частности, такой тест проверяет, обрабатывает ли проверяемый автомат входные последовательности не медленнее (не быстрее), чем детерминированный автомат-спецификация [2].

Интуитивно понятно, что сравнение скорости обработки входных воздействий имеет смысл только для автоматов, которые на каждую временную входную последовательность выдают одинаковые выходные последовательности. Такие автоматы будем называть *f-эквивалентными* [7, 8].

Если проверяемый автомат имеет число состояний, равное числу состояний автомата-спецификации, т.е. принадлежит множеству $\mathfrak{I}_n(I, O, B)$, $n = |S|$, то для проверки соответствия времени реакции на входные символы достаточно построить обход графа переходов автомата S .

Для каждого *ir*-состояния s автомата S , строим последовательности $\alpha_s \cdot (i, 0)$, $\alpha_s \in V$, $i \in I$. Объединяем все последовательности во множество TTS_1 . Для каждого *iur*-состояния s' , находим пару $\upsilon(s') = \{\alpha_s, \tau\}$ и строим последовательности $\alpha_s \cdot (i, \tau)$, $i \in I$. Объединяем все последовательности во множество TTS_2 .

Множество $TTS = TTS_1 \cup TTS_2$ есть полный тест для проверки, является ли автомат P строго не медленнее (строго не быстрее) автомата S по времени обработки временных входных последовательностей.

Утверждение 4. Если автомат $P \in \mathfrak{I}_n(I, O, B)$ является *f-эквивалентным* автомату S , то автомат P строго не медленнее (строго не быстрее) автомата S , если и только если время обработки каждого временного входного символа из последовательностей множества TTS не больше (соответственно не меньше) времени обработки этого символа автоматом S .

Заметим, что методы построения полного проверяющего теста, предложенные в главе 3, включают в себя тест для проверки *f-эквивалентности*, а также включают в себя последовательности теста, необходимые для проверки соответствия времени обработки входных символов.

В **заключении** формулируются основные результаты работы.

1. Предложен метод построения по временному автомату соответствующего конечного автомата, который позволяет строить проверяющие тесты с гарантированной полнотой для временных автоматов на основе известных методов из классической теории автоматов.

2. Установлены необходимые и достаточные условия эквивалентности двух временных автоматов на основе соответствия между конечными множествами их состояний. Данное соответствие позволяет адаптировать методы построения проверяющих тестов относительно модели «черного ящика» из классической теории автоматов для построения проверяющего теста непосредственно по модели временного автомата, поскольку тесты, построенные по соответствующему классическому конечному автомату, обладают значительной избыточностью.

3. Предложен метод построения полного проверяющего теста для временного автомата относительно модели «черного ящика», на его основе разработаны алгоритмы для построения полных проверяющих тестов относительно различных модификаций модели «черного ящика».

Отметим, что длину полного проверяющего теста относительно модели «черного ящика», построенного по одному из предложенных алгоритмов, можно уменьшить как за счет применения более эффективных методов использования различающих последовательностей, как это делается в *HSI*, *H* и *SPY* методах для классических конечных автоматов, так и за счет сокращения области неисправности, как предлагается в главе 4.

В приложении А приведены результаты синтеза полных проверяющих тестов для случайно сгенерированных автоматов. Приложение Б содержит расширенный автомат для протокола SCP и график зависимости полноты теста от длины теста при пассивном тестировании реализаций протокола SCP. В приложении В приведены копии документов о внедрении результатов работы в учебный процесс.

СПИСОК ПУБЛИКАЦИЙ ПО ТЕМЕ ДИССЕРТАЦИИ

1. Жигулин М.В. Оценка полноты проверки при пассивном тестировании на основе автоматной модели / М.В. Жигулин, А.В. Коломеец // Известия Томского политехнического университета. – 2009. – Т. 314, № 5. – С. 225-228.

2. Жигулин М.В. Синтез тестов с гарантированной полнотой для временных автоматов / М.В. Жигулин, И.М. Дмитриев, Н.В. Евтушенко // Известия Томского политехнического университета. – 2010. – Т. 316, № 5. – С. 104-110.

3. Дмитриев И.М. Синтез проверяющих тестов с гарантированной полнотой для временных автоматов с конечными задержками / И.М. Дмитриев, М.В. Жигулин // Известия высших учебных заведений. Физика. – 2010. – Т. 53, № 9/3. – С. 196-198.

4. М.В. Жигулин Тестирование программной реализации протокола IRC на основе модели расширенного автомата / М.В. Жигулин, А.В. Коломеец, Н.Г. Кушик, А.В. Шабалдин // Известия Томского политехнического университета. – 2011. – Т. 318, № 5. – С. 81-84.

5. Жигулин М.В. Преобразование спецификации при пассивном тестировании телекоммуникационных протоколов / М.В. Жигулин, С.А. Прокопенко, Н.В. Евтушенко // Новые информационные технологии в исследовании сложных структур : доклады VI Всероссийской конференции с международным участием // Вестник Томского государственного университета. Приложение. – Томск, 2006. – № 18. – С. 62-66.

6. Жигулин М.В. Оценка полноты проверки при пассивном тестировании на основе автоматной модели / М.В. Жигулин // Новые информационные технологии в исследовании сложных структур : тезисы докладов седьмой Российской конференции с международным участием. – Томск, 2008. – С. 48.

7. Жигулин М.В. Необходимое и достаточное условие f -эквивалентности временных автоматов / М.В. Жигулин, И.М. Дмитриев // VI Всесибирский конгресс женщин-математиков : материалы Всероссийской конференции. – Красноярск, 2010. – С. 137-141.

8. Проведение прикладных и проблемно-поисковых исследований в области информационно-телекоммуникационных систем с участием научных организаций Франции. Этап 4. Экспериментальные исследования поставленных перед НИР задач. Обобщение и оценка результатов исследования: отчет о НИР (заключительный) / Том. гос. ун-т ; рук. Н.В. Евтушенко. – Томск, 2010. – 301 с. – № ГР 01200962727.

9. Жигулин М.В. Алгоритм синтеза проверяющего теста с гарантированной полнотой для конечных временных автоматов с фиксированным набором задержек для модели «черного ящика» / М.В. Жигулин, И.М. Дмитриев, Д.Д. Попов // Новые информационные технологии в исследовании сложных структур : тезисы докладов восьмой Российской конференции с международным участием. – Томск, 2010. – С. 50.

10. Zhigulin M. FSM-Based Test Derivation Strategies for Systems with Time-Outs / M. Zhigulin, N. Yevtushenko, S. Maag, A. Cavalli // 11th International conference on quality software. – Madrid, 2011. – P. 141-150.