Московский Государственный Университет
имени М.В. Ломоносова
Факультет Вычислительной Математики и Кибернетики

На правах рукописи

Коваленко Дмитрий Сергеевич

МЕТОДЫ И ПРОГРАММНЫЕ СРЕДСТВА ОБУЧЕНИЯ АЛГОРИТМОВ РАСПОЗНАВАНИЯ УЧАСТКОВ ФАЗОВЫХ ТРАЕКТОРИЙ

Специальность 05.13.11 — математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей

АВТОРЕФЕРАТ
диссертации на соискание ученой степени
кандидата физико-математических наук

Москва
2011

Работа выполнена на кафедре Автоматизации Систем Вычислительных Комплексов факультета Вычислительной Математики и Кибернетики Московского Государственного Университета имени М.В. Ломоносова.

Научный руководитель:	кандидат технических наук, старший научный сотрудник Костенко Валерий Алексеевич
Официальные оппоненты:	доктор физико-математических наук, профессор, член-корреспондент РАН Рудаков Константин Владимирович
	кандидат физико-математических наук, старший научный сотрудник Литинский Леонид Борисович
Ведущая организация:	ОАО «НИИ ВК им. М.А. Карцева»

Защита состоится « 4 » марта 2011 г. в 11:00 на заседании диссертационного совета Д 501.001.44 при Московском Государственном Университете имени М.В. Ломоносова по адресу: 119991, ГСП-1, Москва, Ленинские горы, МГУ, 2-ой учебный корпус, факультет Вычислительной Математики и Кибернетики, аудитория 685.

С диссертацией можно ознакомиться в библиотеке факультета ВМиК МГУ имени М.В. Ломоносова, с текстом автореферата — на официальном сайте ВМиК МГУ имени М.В. Ломоносова: `http://www.cs.msu.su` в разделе «Наука» — «Работа диссертационных советов» — «Д 501.001.44».

Автореферат разослан « 2 » февраля 2011 г.

Ученый секретарь
диссертационного совета Д 501.001.44

профессор Н.П. Трифонов

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность темы. В настоящее время существует множество сложных систем, которые требуют автоматического контроля своего состояния и действий. В работе рассматриваются системы, поведение которых описывается фазовыми траекториями, получаемыми с датчиков, окружающих систему. К классу таких систем относятся биологические, технические и другие виды систем. Одним из способов контроля состояния и действий системы является анализ фазовых траекторий.

Если для некоторой системы известны возможные типы нештатного поведения[1], и для каждого из типов известна характерная фазовая траектория, то возникает задача распознавания нештатного поведения системы по показаниям датчиков. От качества решения этой задачи распознавания может зависеть как безопасность техники и персонала, так и экономическая эффективность эксплуатации системы. При этом алгоритм распознавания должен работать в реальном времени.

Для ряда систем построение алгоритмов распознавания с использованием моделей системы, основанных на происходящих в ней физических процессах, невозможно. В этом случае применяют алгоритмы, основанные на методах машинного обучения. Для построения таких алгоритмов используется обучающая выборка. Эта выборка состоит из фазовых траекторий, полученных путем моделирования работы системы или путем наблюдения за реальной системой, для которых указаны участки нештатного поведения.

Одной из проблем применения существующих методов являются искажения траекторий, характерных нештатному поведению рассматриваемой системы, в различных условиях работы. Проведенные

[1] Последовательности действий или событий, происходящих в рассматриваемой системе и приводящих ее в состояние, в котором она не выполняет своих функций.

исследования показали, что существующие методы распознавания не обладают одновременно устойчивостью к искажениям траекторий нештатного поведения по амплитуде и времени. В связи с этим, актуальной является задача построения семейства алгоритмов распознавания и разработки методов его обучения по обучающей выборке, которые позволят получать алгоритмы распознавания, устойчивые к искажениям фазовых траекторий по амплитуде и времени.

Существует ряд систем, для которых известными являются только моменты времени, когда система уже находится в аварийном состоянии, а границы предшествующих участков нештатного поведения определить не удается. В этом случае для траекторий обучающей выборки известны только точки наступления аварийных состояний. Существующие методы обучения алгоритмов распознавания не приспособлены для работы с обучающей выборкой, для траекторий которой заданы только точки наступления аварийных состояний, а границы участков нештатного поведения неизвестны. Актуальной является разработка методов обучения по выборке с указанием точек аварий для предложенного семейства алгоритмов, которые позволят получать устойчивые к искажениям фазовых траекторий алгоритмы распознавания.

Цель работы. Целью данной работы является разработка методов и программных средств обучения алгоритмов распознавания нештатного поведения систем по обучающей выборке, которые обладают высокой устойчивостью к искажениям фазовых траекторий системы, а также определение условий, при которых возможно применение разработанных методов. Для достижения цели необходимо решить следующие задачи:

- Разработать параметрическое семейство алгоритмов распознавания нештатного поведения систем, поведение которых описывается фазовыми траекториями, и получить условия его обучаемости.

- Для предложенного семейства разработать алгоритмы обучения и разработать алгоритм их распараллеливания для кластерных многоядерных вычислительных систем.
- На основе предложенных алгоритмов разработать программные средства обучения алгоритмов распознавания и средства исследования устойчивости алгоритмов к искажениям траекторий по амплитуде и времени.
- Провести исследования предложенного метода и существующих методов на модельных данных с целью определения их устойчивости к искажениям фазовых траекторий системы и провести апробацию разработанных средств для решения практических задач.

Методы исследования. Для достижения указанных целей в работе были использованы методы математического программирования, методы обучения по прецедентам, методы цифровой обработки сигналов и алгоритмы интеллектуального анализа данных.

Основные результаты работы. Результаты диссертационной работы заключаются в следующем:

1. Предложено и исследовано ориентированное на параллельную реализацию параметрическое семейство алгоритмов распознавания нештатных режимов работы систем, поведение которых описывается фазовыми траекториями. Алгоритмы из данного семейства обладают высокой устойчивостью к искажениям фазовых траекторий нештатных режимов работы по сравнению с известными алгоритмами.
2. Для предложенного семейства разработаны оригинальные алгоритмы обучения и алгоритмы их распараллеливания для кластерных многоядерных вычислительных систем.
3. На основе предложенных алгоритмов разработаны инструментальные средства автоматического построения и исследования свойств

алгоритмов распознавания. Разработанные средства прошли апробацию при решении практических задач.

Научная новизна. В диссертации представлено новое параметрическое семейство алгоритмов распознавания нештатного поведения систем, поведение которых описывается фазовыми траекториями. Алгоритмы из данного семейства основаны на идеях алгебраического подхода к разметке плоских конфигураций. Сформулировано и доказано необходимое условие обучаемости предложенного семейства.

Разработан новый метод обучения алгоритмов распознавания. Для случая задания обучающей выборки с указанием участков нештатного поведения разработан новый генетический алгоритм построения системы аксиом, который на каждой итерации производит автоматическое определение значений параметров операций мутации и скрещивания на основе анализа текущей популяции. Данный алгоритм обладает более высокой точностью по сравнению с существовавшим ранее генетическим алгоритмом построения системы аксиом. Для случая задания обучающей выборки с указанием только точек аварии разработан новый алгоритм построения системы аксиом, который позволяет формировать системы аксиом и эталонные траектории.

Практическая ценность. В работе показано, что алгоритмы из разработанного параметрического семейства алгоритмов распознавания обладают лучшей устойчивостью к наложению шумов и искажениям траекторий нештатного поведения по амплитуде и времени по сравнению с существующими алгоритмами. Созданы инструментальные средства, реализующие все предложенные алгоритмы обучения, которые допускают распараллеливание на кластерных многоядерных вычислительных системах. Результаты данной работы могут быть использованы при проектировании и эксплуатации сложных систем, требующих автоматического контроля своего

состояния.

Расширена область применения методов автоматического распознавания нештатного поведения сложных систем на случай, когда для траекторий обучающей выборки заданы только точки аварий. Это позволяет создавать алгоритмы распознавания нештатного поведения систем, для которых невозможно однозначно указать участки нештатного поведения, предшествующие точкам наступления аварийных ситуаций.

Апробация работы. Результаты, представленные в работе, докладывались на научном семинаре Лаборатории Вычислительных Комплексов кафедры АСВК факультета ВМиК МГУ имени М.В. Ломоносова под руководством профессора Р.Л. Смелянского; на семинаре кафедры АСВК под руководством заведующего кафедрой член-корр. РАН Л.Н. Королева; а также на следующих конференциях:

- II Всероссийская научная конференция «Методы и средства обработки информации», Москва, 2005;
- VII Международная конференция «Дискретные модели в теории управляющих систем», Москва, 2006;
- VI Международная конференция «Интеллектуализация обработки информации», Алушта, Украина, 2006;
- V Международная конференция по исследованию операций «ORM», Москва, 2007;
- VII Международная конференция «Интеллектуализация обработки информации», Алушта, Украина, 2008;
- XI Всероссийская конференция «Нейроинформатика», Москва, 2009;
- III Всероссийская научная конференция «Методы и средства обработки информации», Москва, 2009;
- V Международная конференция «IEEE BIC-TA», Чанша, Китай, 2010.

Публикации. По теме диссертации опубликовано 8 печатных работ, в

том числе работа в журнале «Моделирование и Анализ Информационных Систем», входящем в перечень рецензируемых научных журналов ВАК РФ. Список работ приведен в конце автореферата.

Структура и объем диссертации. Диссертация состоит из введения, шести глав, заключения, списка литературы и трех приложений. Объём работы — 146 страниц, объём приложений составляет 22 страницы. Список литературы содержит 62 наименования.

ОСНОВНОЕ СОДЕРЖАНИЕ РАБОТЫ

Во **введении** обоснована актуальность диссертационной работы и сформулирована цель исследований.

В **первой главе** описан класс рассматриваемых систем и сформулированы задачи распознавания нештатного поведения сложных систем по показаниям датчиков.

Рассматриваются системы, информация о поведении которых доступна в виде фазовых траекторий, получаемых с окружающих систему датчиков. Время рассматривается дискретное, датчики опрашиваются с некоторой фиксированной частотой. Состояние системы характеризуется точкой фазовой траектории. Со временем система изменяет свое состояние, последовательные изменения состояния системы будем называть ее поведением. Оно характеризуется фазовой траекторией. Существует три класса состояний системы:

1. Штатное состояние, при котором система стабильно выполняет заложенные в нее функции.
2. Нештатное состояние, при котором система в скором времени гарантированно перестанет выполнять заложенные в нее функции.
3. Аварийное состояние, при котором система не выполняет заложенных в нее функций.

Нормальным поведением системы будем называть такое поведение, при котором все состояния, которые принимает система, принадлежат классу штатных состояний. Нештатным поведением системы будем называть такое, при котором все состояния, которые принимает система, принадлежат классу нештатных состояний. Может существовать несколько классов нештатного поведения, приводящих к аварийному состоянию системы. Классы нештатного поведения определяются различными неисправностями, которые могут возникнуть в рассматриваемой системе. Для каждого класса нештатного поведения существует некоторая характерная фазовая траектория X_{Anom}, такие траектории будем называть эталонными.

Участки нештатного поведения могут входить в анализируемую фазовую траекторию X в искаженном относительно эталонных траекторий виде. Можно выделить следующие типы искажений: искажения по амплитуде и искажения по времени. Подклассом искажений по амплитуде можно считать стационарный шум, возникающий в датчиках, окружающих систему.

Задача распознавания нештатного поведения систем, информация о которых доступна в виде фазовых траекторий, сформулирована следующим образом. Дано:

- Набор классов нештатного поведения системы, для каждого из которых задана эталонная траектория X^l_{Anom}.
- Наблюдаемая фазовая траектория X, содержащая участки нештатного поведения, которые могут быть искажены относительно эталонных траекторий.
- Ограничения на полноту и точность распознавания:

$$e_1 \leq c_1 \text{ и } e_2 \leq c_2 \tag{1}$$

где: e_1, e_2 — число ошибок распознавания I и II рода соответственно; c_1, c_2 — заданные числовые ограничения.

- Ограничение на время работы алгоритма распознавания:

$$t_{Al(X)} \leq c_3 \tag{2}$$

где: $t_{Al(X)}$ – время работы алгоритма распознавания Al над траекторией X; c_3 – заданное ограничение, которое определяется в зависимости от скорости развития процессов в наблюдаемой системе.

Требуется провести распознавание нештатного поведения в работе наблюдаемой системы с учетом ограничений на время работы, полноту и точность распознавания.

Ключевой особенностью рассматриваемых систем, которая затрудняет решение задачи распознавания, является наличие шумов и нелинейных искажений по амплитуде и времени участков нештатного поведения.

Рассмотрим задачу обучения алгоритмов распознавания. Информация о системе доступна в виде набора прецедентов TS, т.е. набора траекторий с указанием участков нештатного поведения. Прецеденты могут быть получены путем наблюдения за реальной системой или путем моделирования. Всю выборку TS можно разделить на обучающую $\widetilde{TS}$ и контрольную выборку $\widehat{TS}$.

Пусть определена целевая функция $\varphi(e_1, e_2)$, где e_1 и e_2 — число ошибок распознавания I и II рода, которая отвечает следующим требованиям:

- Функция $\varphi(e_1, e_2)$ определена для любых неотрицательных целых значений аргументов e_1 и e_2.
- Функция $\varphi(e_1, e_2)$ монотонно не убывает по каждому из аргументов.

Задача обучения алгоритмов распознавания формулируется в виде задачи обучения по прецедентам. Дано:

- Обучающая выборка $\widetilde{TS}$ и контрольная выборка $\widehat{TS}$,
- Целевая функция $\varphi(e_1, e_2)$.

Требуется построить алгоритм распознавания Al, который будет удовлетворять следующему набору ограничений:

1. Алгоритм Al должен выдавать ограниченное число ошибок на обучающей выборке $\widetilde{TS}$:

$$e_1(Al, \widetilde{TS}) \leq P^1_{wrong}, e_2(Al, \widetilde{TS}) \leq P^2_{wrong} \quad (3)$$

где: P^1_{wrong} и P^2_{wrong} — заданные числовые ограничения.

2. Алгоритм Al должен минимизировать значение целевой функции $\varphi(e_1, e_2)$ на контрольной выборке $\widehat{TS}$:

$$Al = arg \min_{Al}(\varphi(e_1(Al, \widehat{TS}), e_2(Al, \widehat{TS}))) \quad (4)$$

3. Вычислительная сложность работы алгоритма распознавания $\Theta_{Al}(m)$ над любой фазовой траекторией, длины не больше m, должна быть ограничена наперед заданной функцией $\theta(m)$, которая определяется характеристиками используемого вычислителя и скоростью развития процессов в анализируемой системе:

$$\Theta_{Al}(m) \leq \theta(m) \quad (5)$$

Для многих систем при формировании выборки TS не всегда возможно однозначно указать участки нештатного поведения в траекториях из этой выборки. При этом известны только точки наступления аварийных состояний, идущие после участка нештатного поведения. Т.е. выборка TS представляет собой множество траекторий, содержащих участки нештатного поведения, границы которых неизвестны, а указаны только точки аварий. Для этого случая задача обучения формулируется аналогично выше приведенной задаче.

Во **второй главе** приведено численное исследование наиболее распространенных алгоритмов решения задачи распознавания заданных эталонных траекторий в наблюдаемой траектории, основанных на методах машинного обучения:

- Алгоритмы на основе искусственных нейронных сетей.

- Алгоритмы, основанные на преобразованиях Фурье и вейвлетах.
- Алгоритм "Гусеница" – Singular Spectrum Analysis (SSA).
- Алгоритмы на основе Dynamic Time Warping (DTW).
- Алгоритмы распознавания на основе идей алгебраического подхода.

Целью исследования методов является сравнение качества их работы в условиях, когда участки наблюдаемой траектории, соответствующие нештатному поведению, искажены относительно эталонных траекторий. Для сравнения методов выбраны следующие критерии: *a*) Количество ошибок распознавания I рода. *b*) Количество ошибок распознавания II рода. *c*) Вычислительная сложность работы алгоритма распознавания.

В *разделах 2.2–2.8* приведены описания каждого из рассматриваемых параметрических семейств алгоритмов распознавания, приведены методы решения задачи обучения алгоритмов распознавания по набору прецедентов. Для каждого из рассматриваемых алгоритмов приведены результаты численного исследования и сделаны выводы по возможности их применения. Для проведения численного исследования был создан программный стенд, позволяющий создавать многомерные траектории с заданными характеристиками искажений и шумов.

В результате проведенного исследования было выявлено, что существующие алгоритмы распознавания недостаточно хорошо справляются с искажениями наблюдаемой траектории по амплитуде и времени. Лучшую устойчивость к искажениям по времени показал алгоритм распознавания, основанный на идеях алгебраического подхода.

Ключевым элементом алгоритмов распознавания, основанных на идеях алгебраического подхода, является разметка фазовой траектории условиями. Эти условия называются аксиомами[2]. Точка траектории размечается аксиомой, если для данной точки условия аксиомы выполняются. Аксиома

[2] Простейшими примерами аксиом являются: условие локального max (или min) значения размечамой точки фазовой траектории, условия возрастания и убывания.

определена на некоторой окрестности размечаемой точки. Системой аксиом называется такой набор аксиом, что любая точка произвольной фазовой траектории может быть размечена только одной аксиомой из данного набора. Разметкой фазовой траектории называется последовательность аксиом такая, что каждой точке траектории соответствует одна аксиома из заданной системы. Распознавание нештатного поведения системы ведется путем поиска разметок эталонных траекторий в разметке анализируемой траектории.

Существующие алгоритмы обучения распознавателей на основе данного подхода обладают рядом недостатков: *a*) высокая вычислительная сложность алгоритмов обучения; *b*) необходимость участия эксперта при построении системы аксиом; *c*) невысокая устойчивость алгоритмов распознавания к наложению шумов; *d*) необходимость повышения точности работы получаемых в результате обучения алгоритмов распознавания в ряде случаев; *e*) отсутствие алгоритмов обучения в случае, когда для траекторий выборки TS указаны точки аварий.

В связи с этим актуальна задача разработки семейства алгоритмов распознавания на основе идей алгебраического подхода и разработка новых алгоритмов обучения для каждого из способов задания выборки TS.

В **третьей главе** описано разработанное параметрическое семейство алгоритмов распознавания и условия его обучаемости.

Определение 1. *Элементарное условие* $ec = ec(x_t^*, t, P)$ *— это функция, определенная на отчете* t *и некоторой его окрестности* x_t^* *на траектории* X*, зависящая от набора параметров* P*, которая принимает значения из множества* $\{true, false\}$.

Определение 2. *Аксиома* $a = a(x_t^*, t)$ *— это булева функция, заданная в виде формулы от элементарных условий, определенных на отсчете* t *и некоторой его окрестности* x_t^* *на траектории* X*:*

$$a(x_t^*, t) = \bigvee_i \bigwedge_j ec_{ij}(x_t^*, t, P_{ij})^{\delta_{ij}} \tag{6}$$

где:

$$ec(x_t^*, t, P)^\delta = \begin{cases} ec(x_t^*, t, P), \text{ при } \delta = 0 \\ \overline{ec}(x_t^*, t, P), \text{ при } \delta = 1 \end{cases}$$

Определение 3. *Конечное множество аксиом* $as = \{a_1, a_2, \ldots, a_M\}$ *будем называть системой аксиом, если оно удовлетворяет условию:*

$$\forall X \ \forall x_t \in X \ \exists! \ a_i \in as : \ a_i(x_t^*, t) = true$$

т.е. для любой точки x_t *произвольной фазовой траектории* X *найдется и притом только одна аксиома* a_i *из множества* as*, условия которой выполняются в данной точке.*

Далее будем считать, что аксиомы в системе аксиом as пронумерованы последовательными натуральными числами: $as = \{a_1, a_2, \ldots, a_M\}$.

Любое множество аксиом возможно представить в виде системы аксиом выполнив следующее:

1. Введение порядка на множестве аксиом. Будем считать что, если аксиома с индексом i выполняется в некоторой точке произвольной фазовой траектории, то никакая аксиома с индексом $j : \ j > i$ не выполняется в данной точке траектории.
2. Добавление тождественной аксиомы a_∞ с наименьшим приоритетом:

   $$as = \{a_1, a_2, \ldots, a_M, a_\infty\}$$

   Тождественная аксиома a_∞ — это аксиома, которая выполняется в любой точке любой фазовой траектории.

Определение 4. *Разметкой траектории* $X = (x_1, x_2, .., x_n)$ *относительно системы аксиом* as *будем называть последовательность* $J = (j_1, j_2, .., j_n)$*, где* j_t *– индекс аксиомы в системе аксиом* as*, условия которой выполняются на отсчете* t *траектории* X*.*

Алгоритм, который сопоставляет траектории X ее разметку относительно заданной системы аксиом, будем называть алгоритмом разметки и обозначать: Λ.

Параметрическое семейство алгоритмов распознавания S — это множество таких алгоритмов Al, каждый из которых включает: a) алгоритм предобработки траекторий $h(X, P_h)$, где: P_h — параметры алгоритма h; b) алгоритм разметки Λ и система аксиом as; c) алгоритм поиска разметок эталонных траекторий $\{J_{Anom}\}$ в разметке J наблюдаемой фазовой траектории $g(J, \{J_{Anom}\}, P_g)$, где: P_g — параметры алгоритма g.

Под алгоритмом предобработки $h(X, P_h)$ будем понимать алгоритм, который реализует функцию преобразования траекторий, ставящую произвольной фазовой траектории в однозначное соответствие некоторую траекторию: $h(X, P_h) : X \longrightarrow \widehat{X}$, где: X — исходная фазовая траектория; $\widehat{X}$ — траектория, полученная из X в результате преобразования. Примерами алгоритмов предобработки являются: алгоритм сглаживания траекторий, алгоритм сжатия или растяжения траектории на заданный коэффициент.

Каждый алгоритм распознавания $Al \in S$ работает следующим образом. При помощи алгоритма $h(X, P_h)$ происходит предобработка эталонных траекторий $\{X_{Anom}\}$ и анализируемой фазовой траектории X. Затем, при помощи алгоритма Λ формируются разметки эталонных траекторий $\{J_{Anom}\}$ и разметка анализируемой траектории J относительно системы аксиом as. После этого, происходит поиск разметок $\{J_{Anom}\}$ в разметке J, на основании результатов которого производится вывод о том, к какому классу относится поведение анализируемой системы на каждом отсчете траектории X.

Алгоритм предобработки включен в состав алгоритма распознавания с целью уменьшения влияния шумовых составляющих наблюдаемой траектории на качество распознавания. Использование алгоритмов нечеткого поиска разметок позволяет бороться с искажениями по времени.

В *разделе 3.2* приведены условия обучаемости параметрического семейства алгоритмов распознавания S.

Определение 5. *Пусть даны две траектории X и Y, $len(X) \leq len(Y)$, и множество элементарных условий $\{ec\}$. Будем говорить, что траектории X и Y различимы в элементарных условиях из множества $\{ec\}$, если:*

$$\forall r \in [1, len(Y) - len(X) + 1] \; \exists t \in [1, len(X)] \,, \; \exists ec_k \in \{ec\} \,, \; \exists P_{ec_k} : \\ ec_k(Y_{r+t}, r + t, P_{ec_k}) \neq ec_k(X_t, t, P_{ec_k}) \tag{7}$$

Различимость траекторий X и Y будем обозначать: $X \overset{\{ec\}}{\nleftrightarrow} Y$.

Необходимое условие обучаемости семейства S для заданной выборки TS сформулировано в виде следующей теоремы.

Теорема 1. *Для того, чтобы в параметрическом семействе алгоритмов распознавания S существовал алгоритм $Al \in S$, решающий с наперед заданной точностью*[3] *основную задачу распознавания нештатного поведения для произвольной наблюдаемой траектории X необходимо, чтобы любые два участка нештатного поведения из выборки TS, соответствующие различным классам нештатного поведения, были различимы в элементарных условиях из множества $\{ec\}$:*

$$\forall \; Y^i_{Anom}, Y^j_{Anom} \in TS, \; w^i \neq w^j : \; Y^i_{Anom} \overset{\{ec\}}{\nleftrightarrow} Y^j_{Anom} \tag{8}$$

где: $\{ec\}$ — множество всех используемых в S элементарных условий; w^i — класс нештатного поведения участка Y^i_{Anom} из выборки TS.

В *разделе 3.2* приведен алгоритм проверки выполнения условий данной теоремы и указана его вычислительная сложность[4].

Четвертая глава посвящена разработанному методу обучения алгоритмов распознавания нештатного поведения сложных систем, поведение которых описывается фазовыми траекториями.

[3] Под точностью понимаются ограничения на число ошибок распознавания I и II рода.

[4] Вычислительная сложность алгоритма проверки квадратично зависит от числа траекторий нештатного поведения в выборке TS и линейно от числа элементарных условий в семействе S.

В *разделе 4.1* описана общая схема разработанного алгоритма обучения:

1. Разбиение всего параметрического семейства алгоритмов распознавания S на подсемейства: $S = \{S_1, S_2, \ldots, S_N\}$. При этом, в каждом подсемействе S_k определен только один тип алгоритма предобработки и тип алгоритма поиска разметок.
2. Для каждого подсемейства S_k строится алгоритм распознавания:
   а. Построение системы аксиом as.
   б. Определение значений P_g для алгоритма поиска разметок g.
   в. Определение значений параметров P_h для алгоритма предобработки исходных данных h.
   г. Проверка критерия останова и переход на шаг 3, в случае его выполнения, иначе переход на шаг 2а.
3. Выбор лучшего из алгоритмов распознавания, полученных на предыдущем шаге данного алгоритма.

Рассмотрим шаги данного алгоритма. Для описания способа разбиения семейства алгоритмов S на подсемейства на *шаге 1* введем определение.

Определение 6. *Под шаблоном алгоритма распознавания из параметрического семейства S будем понимать следующую тройку элементов $(h^{\{P_h\}}, \{ec\}, g^{\{P_g\}})$:*

- *заданный алгоритм предобработки h и диапазон допустимых изменений значений его параметров $\{P_h\}$;*
- *фиксированное множество элементарных условий $\{ec\}$ и диапазон допустимых изменений значений параметров каждого из них;*
- *заданный алгоритм поиска разметок g и диапазон допустимых изменений значений его параметров $\{P_g\}$.*

Шаблон $(h^{\{P_h\}}, \{ec\}, g^{\{P_g\}})$ определяет подсемейство алгоритмов распознавания в семействе S. Выделение подсемейств происходит путем разделения S на шаблоны. Для каждой пары: тип алгоритма предобработки

h_i и тип алгоритма поиска разметок g_j – создается шаблон $S_k = (h_i, \{ec\}, g_j)$. Множество элементарных условий $\{ec\}$ в каждом шаблоне совпадает с множеством всех элементарных условий $\{ec\}$, определенных в S. Диапазоны изменения значений параметров условий не ограничиваются. Если число шаблонов необходимо увеличить, то создается заданное число шаблонов с одинаковыми типами алгоритмов предобработки и поиска разметок, при этом разграничивается область изменения их параметров.

Такое разбиение семейства S на подсемейства позволяет снизить сложность обучения по сравнению с обучением без разбиения, так как: *a*) изменение типа алгоритма предобработки может существенно изменить входные данные для алгоритма построения системы аксиом, в рамках одного шаблона тип алгоритма предобработки фиксирован; *b*) изменение типа алгоритма поиска разметок может существенно изменить рельеф целевой функции для алгоритма распознавания в целом, в рамках одного шаблона его тип фиксирован; *c*) область поиска удается ограничить, отсекая подсемейства, алгоритмы распознавания в которых обладают неприемлемой вычислительной сложностью. Кроме того, разбиение S на заданное число шаблонов обеспечивает масштабируемость алгоритма обучения при его параллельной реализации для кластерных многоядерных вычислительных систем.

На *шаге 2а* алгоритма обучения происходит построение системы аксиом. Для случая задания выборки TS с указанием участков нештатного поведения был разработан генетический алгоритм, который описан в *разделе 4.2*. Для случая задания выборки TS с указанием точек аварий был разработан алгоритм, основанный на схеме ограниченного перебора. Его описание приведено в *разделе 4.3*.

На *шагах 2б-2в* используются локально-оптимальные алгоритмы поиска значений параметров алгоритмов предобработки и поиска разметок.

Критериями останова итерационного алгоритма поиска решения в рамках шаблона S_k на *шаге 2г* алгоритма обучения являются: *a*) выполнены условия задачи обучения алгоритма распознавания; *b*) общее число итераций алгоритма превысило заданный параметр; *c*) число итераций без улучшения значения целевой функции $\varphi(e_1, e_2)$ превысило заданный параметр.

Выбор лучшего из построенных алгоритмов распознавания на последнем шаге алгоритма происходит по значению целевой функции $\varphi(e_1, e_2)$ на контрольной выборке $\widehat{TS}$.

В *разделе 4.2* рассматривается алгоритм построения системы аксиом as по выборке TS с указанием участков нештатного поведения. Для построения системы аксиом был разработан алгоритм, основанный на схеме генетического алгоритма. Особью в популяции является система аксиом.

Операции мутации и скрещивания определены на трех уровнях: *a*) уровень элементарных условий; *b*) уровень аксиом; *c*) уровень систем аксиом. Вследствии чего, данные операции обладают большим набором параметров. От выбора значений этих параметров напрямую зависит точность и скорость сходимости генетического алгоритма.

Для учета этой особенности были разработаны функции значимости (для систем аксиом, аксиом и элементарных условий) и разработаны методы автоматического определения на каждой итерации генетического алгоритма значений параметров операций мутации и скрещивания в зависимости от значений этих функций. Функции значимости определяются в зависимости от значений целевой функции $\varphi(e_1, e_2)$ для системы аксиом и от частоты срабатывания элементарных условий и аксиом на траекториях обучающей выборки $\widetilde{TS}$. Автоматическое определение параметров операций мутации и скрещивания позволило улучшить точность и скорость сходимости генетического алгоритма по сравнению со случаем, когда параметры операций задаются вручную перед началом выполнения алгоритма.

В *разделе 4.3* рассматривается алгоритм построения системы аксиом as по выборке TS с указанием точек аварии. Перед применением алгоритма вся выборка TS преобразуется. Каждая траектория $X \in TS$, содержащую точку аварии, разделяется на части: траектория нормального поведения X' и траектория X'' длиной r отсчетов, содержащая участок нештатного поведения (r – параметр преобразования). Траектория X'' представляет собой участок траектории X с отсчета $(max(1, I_X - r))$ по отсчет (I_X), где I_X — это точка аварии на траектории X. После преобразований выборка TS представляет собой набор траекторий: $TS = \{X'\} \cup \{X''\}$. Часть из них является траекториями нормального поведения $\{X'\}$, другая часть $\{X''\}$ – это набор таких траекторий, каждая из которых содержит участок нештатного поведения, однако не известно на каких именно отсчетах.

Разделим всю преобразованную выборку TS на 3 равные непересекающиеся части: выборка $\overline{TS}$ для определения разметок участков нештатного поведения, обучающая выборка $\widetilde{TS}$ и контрольная выборка $\widehat{TS}$.

Алгоритм построения системы аксиом состоит в следующем:

1. Для каждого класса $l \in [1, L]$ нештатного поведения системы производится построение элементарных условий и аксиом, которые выполняются на участках нештатного поведения именно этого класса:
   а. Из множества всех входящих в шаблон S_k элементарных условий $\{ec\}$ путем перебора типов условий и значений их параметров выбираются те, которые чаще выполняются на траекториях, содержащих участки нештатного поведения класса l, и реже на траекториях нормального поведения из обучающей выборки $\widetilde{TS}$.
   б. Формирование множества различных аксиом из выбранных элементарных условий. Аксиомы, которые чаще выполняются на траекториях, содержащих участки нештатного поведения класса l, и реже на траекториях нормального поведения из обучающей

выборки $\widetilde{TS}$, поступают на вход следующему этапу алгоритма.

2. Формирование множества систем аксиом из построенных аксиом. Для каждой системы формируются разметки эталонных траекторий. Эти разметки получаются из выборки $\overline{TS}$ путем выделения наибольшей общей подпоследовательности разметок траекторий, содержащих участки нештатного поведения одного класса, относительно данной системы аксиом. Выбор лучшей системы as осуществляется по значению целевой функции $\phi(e_1, e_2)$ на контрольной выборке $\widehat{TS}$.

Основной особенностью предложенного алгоритма построения системы аксиом является то, что алгоритм позволяет выделять разметки участков нештатного поведения, используемые как разметки эталонных траекторий.

В **пятой главе** рассматривается алгоритм распараллеливания разработанных алгоритмов обучения для кластерных многоядерных вычислительных систем и приводится исследование его эффективности. Этот алгоритм состоит из двух этапов:

1. *Параллельное выполнение алгоритмов обучения в рамках различных шаблонов.* Обучение алгоритмов распознавания в рамках различных шаблонов производится независимо друг от друга. Поэтому обучение в рамках различных шаблонов выделяется в отдельные процессы, которые автоматически распределяются по различным узлам кластера.
2. *Распараллеливание алгоритма обучения в рамках одного шаблона.* Наиболее вычислительно-ёмкими операциями алгоритмов обучения в рамках одного шаблона являются операции вычисления целевых функций (для особей популяции в генетическом алгоритме; для элементарных условий, аксиом и систем аксиом в алгоритме построения системы аксиом по выборке с указанием точек аварии). Их выполнение выделяется в отдельные нити процессов, что позволяет задействовать все вычислительные ядра каждого из узлов.

В *разделе 5.2* описаны составляющие созданной программной библиотеки *AxiomLib*. Для работы с библиотекой *AxiomLib* был создан графический пользовательский интерфейс (ГПИ), который так же описан в *разделе 5.2*. Все разработанные средства являются кросс-платформенными.

Раздел 5.3 посвящен исследованию эффективности предложенного алгоритма распараллеливания разработанных методов обучения. Для исследования эффективности предложенного алгоритма был проведен набор экспериментов на кластере, состоящем из шести SMP-узлов, каждый узел которого содержал два 4–х ядерных процессора. В работе приведены подробные характеристики используемого вычислителя и описаны используемые наборы данных. В среднем ускорение времени работы алгоритма обучения на 6 узлах кластера по сравнению со временем работы на одном узле составило 5.64 раза. Загрузка вычислительных ядер системы при этом составляла $95\% - 99\%$.

Шестая глава посвящена численному исследованию разработанных алгоритмов обучения. Исследования проводились как на модельных, так и на реальных данных. Целью исследования на модельных данных является сравнение качества работы получаемых алгоритмов распознавания с другими известными алгоритмами, рассмотренными в главе 2, и определение влияния искажений по амплитуде и времени, наложения шума на число ошибок распознавания получаемых алгоритмов. Целью исследования алгоритмов обучения на реальных данных является апробация разработанного семейства алгоритмов распознавания и методов его обучения при решении практических задач.

В *разделе 6.1* приводится описание используемых наборов данных. В *разделах 6.2–6.3* описаны полученные результаты исследований и проведено сравнение разработанных алгоритмов с существующими.

Результаты сравнения алгоритма распознавания, построенного при

помощи разработанного метода по обучающей выборке с указанием участков нештатного поведения, с другими алгоритмами, рассмотренными в главе 2, представлены в таблице 1. Результаты показаны для следующих характеристик искажений участков нештатного поведения относительно эталонных траекторий: искажения по времени до 40%, по амплитуде до 2.5%, амплитуда шума до 25% от амплитуды сигнала.

Таблица 1. Результаты работы алгоритмов распознавания на модельных данных.

Тип алгоритма распознавания	Число ошибок I рода	Процент ошибок II рода	Сложность алгоритма распознавания
Нечеткое сопоставление строк	501	33%	$O(\sum_{i=1}^{L} N_i \cdot M)$
Dynamic Time Warping (DTW)	175	31%	$O(\sum_{i=1}^{L} N_i^2 \cdot M)$
Нейросетевой подход	1453	20%	$O(d \cdot M)$ где: d — число нейронов
Преобразования Фурье	323	35%	$O(\sum_{i=1}^{L} N_i \cdot \log N_i \cdot M)$
Вейвлет-преобразования	1136	32%	$O(\sum_{i=1}^{L} N_i^2 \cdot \log N_i \cdot M)$
"Гусеница"-SSA	741	21%	$O(\sum_{i=1}^{L} N_i^2 \cdot M)$
Алгоритм на основе идей алгебраического подхода	146	6%	$O((\sum_{i=1}^{L} N_i + +K_{ec}(as)) \cdot M)$ где: $K_{ec}(as)$ — число элем. условий в as
Разработанный метод	76	2%	$O_{h(X)} + K_{ec}(as) \cdot \widehat{M} + +\widehat{M} \cdot \sum_{i=1}^{N} O_{g(J^i_{Anom})}$ где: $O_{h(X)}$ — сложность алгоритма предобработки

Обозначения в таблице 1: M — длина анализируемой траектории X; $\widehat{M}$ — длина траектории $\widehat{X}$, полученной из X в результате предобработки; N_i — длина X^i_{Anom}; L — число классов нештатного поведения.

В условиях существенных искажений участков нештатного поведения алгоритм распознавания, построенный с помощью предложенного метода,

показал низкое число ошибок распознавания как первого так и второго рода по сравнению с другими исследованными методами. Кроме того, он показал лучшую устойчивость к росту искажений участков нештатного поведения по амплитуде и времени.

Для проведения исследований на реальных данных была рассмотрена задача прогнозирования периодов микросна водителя по показаниям датчиков электроэнцефалограммы (ЭЭГ). Данные представляют собой набор 4–мерных траекторий, отражающих показания датчиков ЭЭГ. Частота дискретизации — 100 Гц. Выделение участков нештатного поведения производилось экспертом на основе поведения испытуемых, движения их глаз, состояние зрачков и т.п. Для каждого испытуемого были получены траектории длиной $2 \cdot 10^6 - 2,5 \cdot 10^6$ отсчетов (что соответствует $6 - 7$ часам наблюдения), число периодов микросна составляло от 12 до 26. Результаты применения разработанного метода: в среднем, процент ошибок II рода составил 5%, а число ложных срабатываний в среднем составило около 15.

В *разделе 6.3* приведено исследование метода обучения алгоритмов распознавания по выборке с указанием только точек аварии. Исследования показали, что предложенный метод позволяет получать алгоритмы распознавания, устойчивые к искажениям участков нештатного поведения по амплитуде и времени относительно эталонных траекторий. При искажениях по времени до 20% и амплитуде до 20% и при амплитуде шума в 5% от амплитуды сигнала были получены следующие результаты: число I первого рода — 51, процент ошибок II рода — 8.5%. При искажениях по времени до 20% и амплитуде до 2.5% и при амплитуде шума в 15% от амплитуды сигнала получены следующие значения ошибок распознавания: 30 ошибок I рода (при длине анализируемой траектории в 30000 отсчетов) и 6% ошибок II рода.

Исследование разработанного метода обучения алгоритмов распознавания по обучающей выборке с указанием точек аварии проводилось

как на модельных, так и на реальных данных. Была рассмотрена задача распознавания рукописных букв по показаниям трех датчиков: движение пера вдоль оси X, оси Y и сила нажатия на перо (ось Z). Частота работы датчиков — 200 Гц. Использованный набор данных содержит 2858 трехмерных траекторий для символов латинского алфавита. Длина каждой траектории составляет от 109 до 205 отсчетов. Результаты применения разработанного метода оказались следующими: число корректно распознанных букв составило 95%, число ложных срабатываний – 58.

В **Приложении А** описаны все элементарные условия, использованные в данной работе для формирования аксиом и систем аксиом. **Приложение Б** содержит описание алгоритмов предобработки фазовых траекторий, реализованных в разработанном программном средстве и использованных в данной работе. **Приложение В** содержит результаты численного исследования, представленные графиками, которые отражают число ошибок распознавания при различных характеристиках искажений для алгоритмов, построенных при помощи существующих методов.

ПУБЛИКАЦИИ ПО ТЕМЕ ДИССЕРТАЦИИ

1. Коваленко Д. С., Костенко В. А., Васин Е. А. Исследование применимости алгебраического подхода к анализу временных рядов. // Труды Второй Всероссийской научной конференции «Методы и средства обработки информации». Издательство факультета ВМиК МГУ, 2005. С. 553-559.
2. Коваленко Д. С., Костенко В. А., Васин Е. А. Генетический алгоритм построения системы аксиом для разметки временных рядов. // Труды VII Международной конференции «Дискретные модели в теории управляющих систем». М.: Макс Пресс, 2006. С. 46-54.

3. Коваленко Д. С., Костенко В. А., Васин Е. А. Автоматическое построение алгоритмов, основанных на алгебраическом подходе, для распознавания предаварийных ситуаций динамических систем. // Искусственный интеллект, № 2. Изд.: Крымский научный центр НАН Украины, Симферополь, 2006. С.130-133.
4. Коваленко Д. С. Методы нечеткого сравнения и голосования для построения распознавателей нештатного поведения динамических систем. // Труды V Московской международной конференции по исследованию операций «ORM-2007». Изд: МАКС Пресс, 2007. С.123-125.
5. Коваленко Д. С., Костенко В. А. Метод построения алгоритмов распознавания, основанных на идеях аксиоматического подхода. // Сборник научных трудов XI Всероссийской научно-технической конференции «Нейроинформатика-2009». Изд: МИФИ, 2009. С. 141-149.
6. Коваленко Д. С. Метод автоматического построения алгоритмов распознавания участков фазовых траекторий. // Моделирование и анализ информационных систем, Т. 16, №4, 2009. Изд.: ЯрГУ. С. 6–21.
7. Коваленко Д. С., Костенко В. А, Барыбин А. К., Тумакин Д. А., Чельдиев М. И. Параллельный алгоритм обучения распознавателей нештатного поведения динамических систем. // Труды Третьей Всероссийской научной конференции «Методы и средства обработки информации». Изд.: МАКС Пресс, 2009. С. 362-373.
8. Kovalenko D., Kostenko V. A Genetic Algorithm for Construction of Recognizers of Anomalies in Behaviour of Dynamical Systems. // Proceedings of the IEEE Fifth International Conference on Bio-Inspired Computing: Theories and Applications, IEEE Press, China, 2010. P. 258-263.