ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

9(193) 2012

ТЕОРЕТИЧЕСКИЙ И ПРИКЛАДНОЙ НАУЧНО-ТЕХНИЧЕСКИЙ ЖУРНАЛ

Издается с ноября 1995 г.

УЧРЕДИТЕЛЬ
Издательство "Новые технологии"

СОДЕРЖАНИЕ

Информация о журнале доступна по сети Internet по адресу **http://novtex.ru/IT.**
Журнал включен в систему Российского индекса научного цитирования.
Журнал входит в Перечень научных журналов, в которых по рекомендации ВАК РФ должны быть опубликованы научные результаты диссертаций на соискание ученой степени доктора и кандидата наук.

УДК 004.056.57

В. А. Васенин, доктор физ.-мат. наук, проф.,
Институт проблем
информационной безопасности,
Московский государственный университет
им. М. В. Ломоносова,
e-mail: vasenin@msu.ru

К созданию международной системы мониторинга и анализа информационного пространства для предотвращения и прекращения военно-политических киберконфликтов

Рассматриваются вопросы, связанные с появлением новой, очень востребованной и многоаспектной задачи создания международной системы мониторинга и анализа информационного пространства для предотвращения и прекращения военно-политических киберконфликтов. Анализируются модели и методы, нормативно-правовые и программные механизмы, которые могут быть положены в основу ее решения. Предлагаются общие подходы, учитывающие уже годами сложившиеся на этом направлении реалии.

Ключевые слова: *информационное пространство, механизмы и анализ, кибербезопасность, военно-политический конфликт*

Исходные посылки и актуальность

В последние 3—5 лет одной из главных тем обсуждения на различных форумах [1—10] является организация совместных, скоординированных действий различных стран на уровне государственных органов, бизнеса и структур гражданского общества в целях предотвращения потенциально возможных или ликвидации уже начавшихся крупномасштабных конфликтов в киберпространстве. Масштабы таких конфликтов определяются высоким уровнем разнопланового (материального, политического, социального и других видов) ущерба от действий противоборствующих в их рамках сторон. Причинами, как правило, являются разногласия различных общественных групп, в том числе представляющих интересы отдельных государств, в вопросах военно-политического, социально-экономического, религиозного характера и ряда других.

Обсуждения и консультации по поводу системы мер и действий, направленных на предотвращение или прекращение крупномасштабных киберконфликтов ведутся на уровне совещаний представителей отдельных государств, в том числе их высшего политического руководства, по дипломатическим каналам, на конференциях и симпозиумах ученых, где предлагаются различные подходы, методы и средства решения возникающих на этом пути задач. Поводом и главным побудительным мотивом активного обсуждения этой темы на международном уровне является перманентно возрастающая заинтересованность различных стран в объединении усилий против угроз подобных конфликтов на криминальной основе, с террористическими целями и угрозы кибервоенных действий. В связи с активным использованием метасети Интернет во всех сферах деятельности человека, перманентным ростом числа размещаемых в этой сети информационно-вычислительных ресурсов и востребованных практикой сервисов, с появлением новых технологий доступа к ним подобные угрозы и способы их реализации во все большей степени приобретают транснациональный характер.

В современном обществе, где наряду с традиционными — материальной, социальной, политической и духовной сферами, развивается тесно связывающая их сфера информационных отношений, эти угрозы становятся все более и более реальными. Вопросы предотвращения подобных угроз в отдельных странах активно изучаются, разрабатываются превентивные меры, методы и средства противодействия проявлениям этих угроз и механизмы оперативного реагирования. Однако таких мер зачастую оказывается недостаточно. Появляется острая необходимость в создании некоторой транснациональной платформы (базового набора понятий, положений, методов и средств) для совместных действий разных стран, которая включала бы общую понятийную основу, общее нормативно-правовое поле, регламентирующее такие действия, необходимую для этого общую техническую и технологическую среду. В ряде стран Европы, в Китае и США проводятся поисковые исследования на этом направлении. Однако судя по материалам, представленным в Интернет, а также по результатам обсуждения на ряде международных форумов, затрагивающих эти вопросы, наибольшее внимание выработке подходов к их разрешению уделяется в

США [20]. Комплексным характером и системным подходом к исследованию среди них выделяются исследования группы ученых из лаборатории компьютерных наук и искусственного интеллекта Массачусетского технологического института (MIT, США). Представительными с этих позиций и претендующими на роль наиболее востребованных практикой в настоящее время позиционируются результаты, представленные в работах [4—10]. Отметим, что именно их вносит американская сторона в качестве базовых (основополагающих) на последних международных форумах, посвященных вопросам киберопасности [3]. В этих работах рассматриваются подходы к созданию моделей, описывающих процессы крупномасштабных конфликтов в мировом киберпространстве. К числу таких конфликтов, в первую очередь, относятся действия с криминальными, террористическими и военно-политическими целями. В качестве киберпространства, элементы которого могут быть использованы как объекты и субъекты (предметы, средства) деструктивного воздействия, в этих исследованиях рассматриваются: метасеть Интернет; сети передачи данных различного назначения; контроллеры критически важных индустриальных систем; другие средства вычислительной техники и автоматизированные комплексы на их основе. В качестве начальных атрибутов для адекватного описания таких деструктивных воздействий рассматриваются субъекты и объекты отношений социально-экономического, политического, военно-технического характера и ряд других. Анализ результатов этих и некоторых других исследований на направлении поиска моделей и методов для построения программных средств противодействия крупномасштабным киберконфликтам стал побудительным мотивом для предложений, выводов и рекомендаций, которые изложены в настоящей публикации.

Опыт, полученный автором в ходе исследования моделей информационных воздействий террористического характера [11, 12], показывает, что построение моделей крупномасштабных конфликтов в киберпространстве при перечисленных выше и используемых в работах [4—10] исходных данных сопряжено с большими сложностями. Они обусловлены огромным числом разноплановых атрибутов и диапазонов принимаемых ими значений, отношений между ними и объектами среды окружения. Эти отношения с необходимостью приходится учитывать при описании семантики процессов, сопровождающих такие конфликты. Заметим, что сами исследователи из MIT именуют такой конгломерат подлежащих учету сущностей "киберслоном". Как следствие, системный анализ исследуемых процессов основан на относительно высокоуровневом представлении их характеристик. В свою очередь, эти характеристики относятся к разным сторонам (аспектам) описания, включая сущности политико-дипломатического и правового, социально-экономического и психологического характера, технического и технологического плана. Представляется естественным именовать такой уровень описания макроуровнем.

Принимая во внимание соображения относительно разных уровней и, соответственно, подходов к описанию моделей процессов противоборства в крупномасштабных киберконфликтах, отметим большой, незаполненный пока содержательными моделями пробел между макроуровнем и уровнем, на котором описываются модели, основанные на традиционных подходах, принятых в безопасности информационных технологий [13—19]. В этой связи представляются актуальными как исследования возможностей уже зарекомендовавших себя подходов, так и поиск новых методов и средств представления моделей, адекватно описывающих процессы, сопровождающие крупномасштабные киберконфликты. Полученные в ходе таких исследований подходы способны заполнить отмеченный разрыв. Настоящая публикация посвящена изложению взглядов автора на направления такого поиска. Они основаны на результатах исследований, в первую очередь, в области противодействия проявлениям кибертеррористической угрозы [11, 12, 28, 29]. Не отрицая возможности использования отмеченных выше макромоделей, далее предлагается подход, который, по мнению автора, является более прагматичным и реализуемым на практике. Он в большей степени ориентирован на уже существующие и "де-факто" используемые на национальных информационных инфраструктурах средства и системы автоматизации процессов контроля (мониторинга) состояния защищенности отдельных национально значимых объектов этих инфраструктур от деструктивных воздействий, на уже приобретенный опыт противодействия реальным кибератакам на подобные объекты.

Анализ различных сторон (аспектов, атрибутов), характеризующих угрозы криминального, террористического, военно-политического характера и способы их реализации, показывают, что между ними существуют отличия. Они связаны с различными целевыми установками, объектами потенциальных атак, возможностями их инициатора по вовлечению ресурсов (людских, материальных, технических и других) в процесс подготовки атаки, ее проведения и, как следствие, с используемыми для этого методами и средствами. Эти различия сказываются на моделях (формальном описании) угроз и процессов их реализации, сопровождающих исследования на этом направлении. Заметим, что в большей степени близки с упомянутых позиций действия террористические и военно-политические, однако и они имеют отличия. В связи с отмеченными обстоятельствами, далее будем рассматривать только крупномасштабные военно-политические киберконфликты.

В качестве примеров последних крупномасштабных компьютерных атак, по используемым средствам и потенциальному ущербу от их реализации соизмеримых с теми, которые могут быть задействованы в военно-политическом киберконфликте с разведывательными целями, являются атаки, выполненные с помощью вредоносных программ (вирусов): Stuxnet — в 2010 г. проведены атаки на компьютеры, обслуживающие ядерные объекты Ирана; Flame — обнаруженный Лабораторией Касперского в 2012 г., сложно организованный многофункционный комплекс программ долговременного действия, ориентированный на компьютеры стран Ближнего Востока. Есть все основания полагать, что заказчиком столь сложных и дорогостоящих комплексов являются структуры, представляющие государство (или государства), имеющие экономические и политические интересы в этом регионе.

Исключаемые еще пять лет назад из рассмотрения на дипломатическом уровне, в первую очередь, по инициативе США [21], в настоящее время военно-политические киберконфликты являются предметом активного изучения и обсуждения на международном уровне. Далее для краткости изложения будем именовать такие конфликты "крупномасштабными" или "военно-политическими", отмечая отдельные характеризующие их аспекты и имея при этом в виду крупномасштабные военно-политические конфликты в информационном пространстве. Отмечая многоаспектный характер подходов к разрешению рассматриваемых вопросов, включая правовые, социально-экономические и политические аспекты, в данной публикации основное внимание обращается на технические и технологические методы и средства их реализации.

1. Методы и средства обеспечения безопасности информационных технологий

Отправными с позиции цели исследований, результаты которых представлены в настоящей работе, являются традиционно используемые методы и средства обеспечения информационной безопасности ресурсов компьютерных систем. Как отмечалось ранее, арсенал таких методов и средств, включая нормативно-правовые, а также программных механизмов, направленных на обеспечение информационной безопасности объектов информационной среды (или информационного пространства), на настоящее время достаточно большой. Эти методы и средства, как правило, основаны:

— на критериальных подходах [13—15, 18, 19] к разработке, внедрению и сопровождению программных средств обеспечения безопасности информационных технологий;

— на формальных, обычно математических, моделях гарантированно защищенных систем [23, 24];

— на методах и средствах верификации (например [25]) проверки соответствия таких программных средств некоторому заданному набору требований (спецификаций), в том числе заданных в виде формальных моделей;

— на тестовых испытаниях (валидации) программных средств обеспечения безопасности в соответствии с заранее принятой программой таких испытаний.

Выбор методов и средств в рамках перечисленных подходов диктуется политикой информационной безопасности подлежащей защите системы. Положения такой политики на верхних уровнях управления безопасностью формируются в виде совокупности общих правил и способов, мер и мероприятий, направленных на безопасное использование информации, технологий и аппаратно-программных средств системы. Исходными для более точного, строгого представления этих положений на нижележащих архитектурных уровнях описания политики безопасности являются формальные модели или требования, описывающие объекты защиты, уязвимости и угрозы их безопасности, потенциальные источники этих угроз и способы реализации. На нижнем уровне управления безопасностью положения политики информационной безопасности специфицируются в виде набора поддерживающих ее методов и средств. Следует, однако, с сожалением констатировать, что осознанный, и как следствие, строго обоснованный выбор таких методов и средств, на практике зачастую игнорируется, что приводит к неэффективному их применению (или не применению вообще), даже по отношению к объектам с повышенными требованиями безопасности. Причиной такого положения, как правило, является отсутствие у участвующих в подобных процессах специалистов (экспертов и администраторов) должных знаний и навыков в области обеспечения информационной безопасности.

Вместе с тем, основным недостатком практической реализации критериальных подходов являются трудности их применения к сложно организованным системам, которые, как правило, и являются главными объектами защиты с их использованием. Выходом здесь является декомпозиция такой системы на отдельные, составляющие ее элементы, применение критериальных подходов к ним с последующей интеграцией методов и средств обеспечения безопасности в рамках сложной системы.

Подходы к разработке средств обеспечения информационной безопасности практически значимых систем на основе их гарантированно защищенных моделей начались с работ, описывающих традиционно применяемые в операционных системах (мониторах безопасности) механизмы дискреционного, мандатного (многоуровневого), а затем и ролевого разграничения доступа. В таких моделях

(Take-Grand, Белла — Лападула, MLS, безопасных информационных потоков и других) нашли отражение особенности широко применяемых политик безопасности, механизмы используемых аппаратно-программных платформ, характеристики среды окружения и ряд других. К недостаткам перечисленных подходов следует также отнести трудности, которые возникают при применении базовых моделей к формальному описанию семантики процессов обеспечения безопасности реальных, сложно организованных объектов.

Подходы на основе проверки соответствия программных средств обеспечения безопасности некоторой, изначально заданной спецификации, в том числе в виде математических моделей, включают более подробно описанные в работе [26] методы:

- методы статического анализа, в числе которых поиск пути в графовой модели, семейство методов "model checking", методы логического программирования и автоматического доказательства теорем, методы формальной верификации императивных программ;
- методы динамического анализа, использующие тестирование и натурное моделирование;
- метод имитационного моделирования, включающий дискретное событийное (*event-driven*) моделирование исследуемой системы.

Вместе с тем, как отмечалось в работе [26], все перечисленные выше методы и, как следствие, соответствующие инструментальные средства имеют недостатки, которые ограничивают возможности их использования на практике. Методы статического анализа и имитационного моделирования в случае применения к сложно организованным объектам представляют большие трудности при их математическом описании. Как правило, возникают вычислительные сложности моделирования и ряд других. Тестовые испытания и натурное моделирование как методы разработки программного обеспечения безопасности подконтрольных ресурсов также имеют свои недостатки. Главные из них:

— отсутствие "полноты покрытия" программы испытаний;

— трудности строгого доказательства перехода от декомпозиции процесса на отдельные его составляющие к его последующему представлению в виде их совокупности (объединения);

— трудности составления и реализации программы испытаний для сложно организованных программных продуктов.

Как следствие недостатков перечисленных выше подходов — сложности сопоставления результатов тестовых испытаний, полученных разными группами исследователей. Последнее обстоятельство уменьшает уровень доверия к таким результатам, сдерживает использование этого программного обеспечения на практике.

Несмотря на все перечисленные недостатки, в процессе разработки и многолетней апробации представленных выше подходов к обеспечению информационной безопасности ресурсов традиционных автоматизированных систем сформировался необходимый для этого понятийный аппарат, широкий спектр методов и средств (включая нормативно-правовые) для решения возникающих на этом направлении задач [14—19]. Все они, как правило, имеют единое понимание и принимаются как инструментарий на транснациональном уровне. В контексте целей настоящей работы необходимо эффективно использовать весь этот арсенал средств и методов для решения задач, возникающих с учетом новых вызовов, в связи с потребностями создания международной системы мер, направленных на предотвращение или прекращение уже начавшихся киберконфликтов военно-политического характера.

Причина в том, что именно на их базе создано "де-факто" используемое на настоящее время в отдельных странах математическое и алгоритмическое обеспечение, а также реализующие его аппаратно-программные средства защиты практически значимых, в том числе критически важных объектов, от деструктивных информационных воздействий. Как следствие, разработка иных подходов и методов, механизмов и программных средств противодействия военно-политической киберугрозе, по мнению автора, бесперспективна на практике. Она не только неэффективна, но и деструктивна, и может на несколько лет задержать начало создания инструментальных средств, реализирующих рассматриваемые в настоящей публикации подходы.

2. К разработке новой платформы: исходные положения

Одной из главных причин необходимости активных действий на рассматриваемом в данной публикации направлении являются объективные трудности формального описания моделей многопараметрических, многовариантных объектов и процессов, сопровождающих военно-политические киберконфликты. Дополнительные трудности разработке таких моделей придает необходимость их использования на транснациональном уровне. На этом уровне процессы взаимодействия субъектов, к сожалению, подчиняются не всегда четко сформулированным положениям международного права. Как следствие, с одной стороны, подлежащие исследованию и разработке модели и методы, реализующие их программные средства защиты таких объектов и процессов от деструктивных воздействий сложно специфицировать, а значит — и надежно верифицировать. С другой стороны, требования к методам и средствам, к их программной реализации, предъявляемые новой, рассматриваемой в данной работе постановкой задачи, существенно повышаются (ужесточаются). Это, соответственно, приводит к необходимости применения более на-

дежных по сравнению с перечисленными выше методами и средствами верификации программных средств обеспечения безопасности. Возникает своего рода "замкнутый круг" противоречащих друг другу условий решения основной в контексте целей настоящей работы задачи, обусловленных новой проблемной областью. Разрешение этих противоречий диктует необходимость иного, более широкого взгляда, а точнее, ревизии, практического анализа и пересмотра базовых положений, определяющих платформу традиционной безопасности информационных технологий, их развития с позиции новых требований.

В связи с изложенными соображениями актуальным становится вопрос о том, какова должна быть новая общая платформа взаимодействия всех заинтересованных сторон, в том числе на транснациональном уровне. Такая платформа, как результат расширения и совершенствования уже существующей базы традиционной безопасности информационных технологий, должна включать:

— понятийную (терминологическую) базу;

— четкие постановки отвечающих современным требованиям задач, решение которых будет направлено на достижение поставленной цели;

— нормативно-правовую и техническую базу для взаимодействия различных заинтересованных структур внутри государства и на транснациональном уровне.

Такая работа возможна только на основе взаимодействия в составе единого коллектива исполнителей, технических специалистов, экспертов-политологов и специалистов в области международного права. Подлежащая разработке целевая платформа должна однозначно (в смысле интерпретации ее положений), точно (в плане семантики) и достаточно полно (с позиции предъявляемых требований) описывать:

— киберпространство, которое потенциально подвержено действиям, именуемым кибервоенными;

— потенциально возможные модели угроз объектам этого киберпространства, модели противоборствующих сторон, которые могут быть вовлечены в рассматриваемый киберконфликт, и сценарии противоборства;

— нормативно-правовые положения (обязательства) сторон, в рамках которых должны осуществляться действия, направленные на предотвращение крупномасштабного, в том числе международного, конфликта.

Основные положения этой платформы должны в смысле математической аналогии с метрическим пространством задавать метрику в подлежащем контролю и анализу киберпространстве. С ее помощью должно быть определено само подконтрольное киберпространство (его периметр), пороговые значения состояния его объектов, за которыми должны следовать те или иные понятные заинтересованным сторонам действия, а также другие его атрибуты и их значения.

Необходимо отметить, что далее при описании киберконфликтов военно-политического характера в основном рассматриваются модели и механизмы деструктивных воздействий и меньше внимания уделяется сценариям и моделям противодействия им. Причина в том, что последние являются производными от первых. Вместе с тем, и это следует отметить, их рассмотрение — предмет отдельного второго этапа исследований, которое во многом определяется результатами, полученными на первом этапе. Его создание является предметом обсуждения в настоящей публикации.

3. Модели формального описания военно-политических киберконфликтов

Представляется целесообразным в качестве макрообъекта — **цели деструктивного кибервоздействия,** подлежащего исследованию, мониторингу и анализу на предмет крупномасштабного киберконфликта, рассматривать **критически важное информационное пространство** и киберинфраструктуру как его составляющую. Необходимо отметить следующие три обстоятельства, которые способствуют такому выбору.

- Критически важное киберпространство, как объект потенциальной кибертеррористической угрозы в разных странах активно изучается уже более 10 лет, систематизируются его атрибуты и арсенал средств обеспечения его информационной безопасности на национальном уровне [11, 12].
- Результаты работ на этом направлении имеют свою национальную специфику и попытки их унификации на транснациональном уровне пока не предпринимались. Это обстоятельство будет существенно затруднять согласование международных мер и действий, в том числе дипломатического характера.
- С одной стороны, некоторые из подлежащих анализу результатов объективно затрагивают вопросы национальной безопасности и не подлежат открытому, в том числе на межгосударственном уровне, обсуждению. С другой стороны, исследование части таких результатов для обеспечения международной безопасности столь же объективно необходимо. Следует заметить, что здесь полная аналогия с исследованиями в других областях, затрагивающих интересы национальной безопасности. Однако практика показывает, что возникающие вопросы разрешаются.
- Вопросы киберконфликтов военно-политического характера изучаются пока мало и результаты такого изучения не согласуются даже с исследованиями в области кибертеррористической безопасности. Вместе с тем необходимо заметить, что активные действия на этом направлении предпринимаются в США, ЕС, в том числе под эгидой НАТО.

С учетом изложенных соображений позиционирование в качестве подлежащего мониторингу и анализу критически важного информационного пространства и составляющих его объектов позволяет использовать полученные ранее в различных странах результаты, включая те, которые относятся к кибертерроризму. Достаточно полная классификация таких объектов, официально принятая в различных странах, представлена в работе [11].

Терминология в области безопасности информационных технологий и защиты информации сложилась уже давно. Перечень основных терминов, позволяющих достаточно полно и строго описывать эту сферу деятельности, исчисляется не сотнями, а тысячами.

Вместе с тем, проводя исследования в области обеспечения кибертеррористической безопасности, мы осознали, что при описании сложно организованных объектов критически важных киберинфраструктур этого запаса не хватает. Часть новых необходимых для такого описания терминов введена и изложена в работах [11, 12]. Некоторое развитие этого словаря терминов в целях создания единой понятийной базы для общения на транснациональном уровне получено в рамках совместных исследований, выполненных коллективом Института Восток-Запад и Института проблем информационной безопасности МГУ им. М. В. Ломоносова [27]. Однако этот набор понятий пока очень ограничен и напрямую не связан с реальными объектами, потенциальными угрозами, сценариями, которые описывают способы их реализации, и с мерами противодействия. Последнее замечание в настоящее время сильно ограничивает возможности объединения усилий на этом направлении, в первую очередь, на международном уровне.

Формирование перечня **моделей потенциально возможных угроз** военно-политического характера и **способов их реализации** — предмет отдельных исследований. При этом предлагается исходить из следующих общих требований:

- в качестве конечных целей реализации этих угроз должны рассматриваться объекты международно принятого (на транснациональном уровне) критически важного киберпространства (например критических киберинфраструктур);
- к числу источников деструктивных воздействий, в первую очередь, следует относить те, которые расположены вне страны, объекты которой потенциально подвержены такому воздействию;
- в качестве возможных деструктивных воздействий следует рассматривать не только информационно-технические, но и воздействия другой природы, в первую очередь, политико-дипломатического характера, а также способствующие реализации конечной цели воздействия с использованием оружия другой природы;
- потенциальный ущерб от реализации угрозы объектам критически важного киберпространства должен превосходить некоторое признанное на национальном или транснациональном уровне пороговое значение;
- пороговые значения ущерба от реализации кибервоенной угрозы, вообще говоря, должны учитывать не только материальные потери и человеческие жертвы, но и политический, социально-экономический, психологический и другие его компоненты.

При формировании модели реализации военно-политической киберугрозы следует иметь в виду ее многоэтапный характер. К числу таких этапов могут относиться:

— подготовительная стадия, включающая, например, рекогносцировку, действия разведывательного плана и другие в составе ее технической составляющей, а также действия по дискредитации страны — потенциального противника на мировой арене;

— начальная стадия, включающая, например, действия технического и политического характера, инициирующие киберконфликт, а также способствующие его усилению, в том числе с вовлечением в конфликт третьих стран;

— основная стадия, когда на первый план выходят технические действия, направленные на причинение противнику максимально возможного многопланового (в отмеченном ранее смысле) ущерба;

— заключительная стадия, на которой предпринимаются меры, оказывающие воздействия информационно-пропагандистского характера на третьи стороны, привлекающие их на сторону агрессора и закрепляющие итоги военно-политического конфликта.

При этом следует принимать во внимание то обстоятельство, что действия на каждом из перечисленных этапов могут иметь и самостоятельное целевое назначение. Примерами использования средств информационного воздействия на отмеченные выше начальном и заключительном этапах могут служить действия НАТО против Югославии (1998—1999 гг.), США в отношении Ирака (2001—2005 гг.), за которыми последовали военные операции, а также интерпретация в Интернет и других СМИ результатов событий, которые сопровождали грузино-абхазский конфликт 2008 г. Последним примером аналогичного характера действий является операция НАТО против Ливии (2011 г.). Судя по сведениям, на настоящее время по такому же сценарию развиваются события в Сирии (2012 г.).

С учетом изложенного, составляющие модели (схемы) на различных этапах реализации угроз включают описание сопровождающих их мер и действий политическо-дипломатического, нормативно-правового, психологического воздействия. Будем именовать их мерами и действиями политического сопровождения основных военно-технических действий.

Целями принятия таких мер и совершения действий являются:

— дискредитация противной стороны в глазах мировой общественности в преддверии действий военно-технического характера и создание таким образом более благоприятных условий, оправдывающих эти действия и позволяющих привлекать другие страны на свою сторону в качестве союзников или сочувствующих;

— снижение уровня психологической готовности потенциального противника к отражению основной фазы крупномасштабной кибератаки;

— усилительно-резонирующий эффект поражающего действия атакующей стороны за счет участия в таких действиях (в тех или иных формах) третьих сторон.

Следует отметить, и это подтверждает мировая практика, что аналогичные политические действия с теми же целями характерны и для моделей (сценариев реализации) военно-политических конфликтов с использованием традиционного оружия физического воздействия, ядерного, химического, бактериологического и других его видов. Это обстоятельство упрощает построение моделей кибератак (реализации военно-политических киберугроз), так как позволяет предусмотреть и использовать свершившиеся ранее, неоднократно обсуждаемые на разных уровнях и понятные мировому сообществу военно-политические действия.

Заметим, что в ходе исследований, направленных на разработку моделей реализации кибервоенных действий, включая действия и атакующей, и защищающейся сторон, некоторые исследователи (специалисты, эксперты) в большей мере уделяют внимание именно действиям военно-политического характера, в ущерб военно-техническим. Примером тому могут служить и ранее упоминавшиеся исследования, проводимые в Массачусетском технологическом институте. Оправданием такому подходу могут быть соображения о том, что действия военно-технического характера и представляющие их модели хорошо исследованы и описаны при изучении вопросов традиционной безопасности информационных технологий. Однако, как было продемонстрировано ранее, такое представление не соответствует действительности, и новые вызовы требуют пересмотра многих положений сложившихся ранее подходов. Как следствие, при построении модели реализации кибервоенных действий представляется целесообразным большее внимание уделять вопросам формирования их военно-технической (технологической) составляющей.

Технологическая составляющая модели киберугроз военно-политического характера соответствует основным классам целей (нарушений), которые эти угрозы преследуют. К их числу в кратком их представлении относятся классы угроз, нарушающих конфидициальность и целостность данных, приводящих к отказу в обслуживании пользователей (DoS, DDoS), зависимость от поставщиков средств вычислительной техники и потеря доверия потенциальных пользователей. Способы реализации деструктивных воздействий могут преследовать (исходить) как угрозы одного из перечисленных классов, так и их различные комбинации. Следует отметить, что выше обозначены традиционные классы угроз в исследованиях кратко представленной ранее безопасности информационных технологий. В контексте целей настоящей публикации их особенности связаны со сложностью объектов критически важного киберпространства как объекта потенциальных атак военно-политического характера, а также предъявляемых к ним требованиям. Потеря доверия к средствам обеспечения безопасности таких объектов может привести к массовой деморализации, снижению уровня психологической готовности к сопротивлению личного состава, обслуживающего такие объекты.

Несмотря на широкий перечень потенциально возможных угроз и способов их реализации, исследования в области их формального описания уже проводятся в США, в странах ЕС, в том числе в рамках проектов НАТО. Результаты некоторых из них были неоднократно представлены на международных конференциях. Анализ потенциально возможных сценариев реализации военно-политических киберугроз позволяет сделать следующие выводы.

- Модели технической составляющей этих сценариев совпадают с аналогичными моделями, которые хорошо изучены в области традиционной безопасности информационных технологий. Различие связано только с особенностью объектов атаки, сложностью их организации и высокими требованиями к ним.
- Модели политической составляющей сценариев не отличаются от аналогичной составляющей, которая присуща (соответствует) сценариям угроз с применением любого другого атакующего оружия. Такие сценарии уже достаточно хорошо изучены и могут быть адекватно описаны.
- Техническая и политическая составляющие сценариев реализации угроз военно-политического конфликта слабо коррелированны и с этих позиций их можно рассматривать и анализировать независимо одна от другой. Этот факт существенно облегчает их формальное описание.

Модели противника (агрессора) в киберконфликтах военно-политического характера необходимо строить (разрабатывать) исходя из того, что нападающей стороной будут государства или представляющие их большие группы людей, которые обладают:

— необходимыми для ведения боевых действий в критически важном киберпространстве техническими средствами, технологиями и специалистами, способными их эффективно (с позиции принятых целеустановок) использовать;

— достаточными экономическими и политическими полномочиями, позволяющими им действовать от имени государства (или имитировать такую деятельность);

— возможностями получения конфиденциальной и/или секретной информации об объектах критически важного киберпространства, об их взаимосвязях (взаимозависимости) между собой, включая уязвимости, которые могут быть использованы для организации деструктивных кибервоздействий против них, и сведения о потенциальном ущербе, который может быть такими действиями нанесен.

Нормативно-правовые положения (обязательства) сторон, в рамках которых можно осуществлять скоординированные действия, направленные на предотвращение еще не начавшихся и/или на прекращение уже начавшихся конфликтов террористического или военно-политического характера, призваны регламентировать:

— деятельность по мониторингу состояния объектов национального критически важного киберпространства на предмет наличия против них деструктивных кибердействий, включая оценку изменения во времени реального и прогнозируемого ущерба;

— сравнение оценок текущего и потенциально возможного ущербов от успешной реализации деструктивных воздействий против объектов критически важного киберпространства с пороговыми значениями и выработка на этой основе рекомендаций для принятия киберконтратакующих и/или боевых атакующих действий, дипломатических мер;

— методы обмена информацией, ее верификации, способы реализации мер и практических действий, направленных на повышение доверия к ним со стороны мирового сообщества;

— практические меры и кибератакующие действия, меры военно-политические (дипломатические), направленные против стороны — инициатора киберконфликта.

Отметим, что разработка и практическая реализация эффективной системы таких нормативно-правовых (регламентирующих) положений и обязательств сторон, в том числе на транснациональном уровне, невозможна без четкого (точно и однозначно трактуемого) критически важного киберпространства, а также моделей нарушителя (атакующей стороны) и потенциальных сценариев киберконфликтов.

Заметим, что здесь вполне уместна аналогия с историей принятия на международном уровне аналогичной системы мер для ограничения стратегических наступательных вооружений или ядерного оружия. В настоящее время такие меры и договоры уже воспринимаются как должное.

Заключение

В настоящей статье в самом общем виде представлены соображения автора по постановке новой и очень актуальной (острой) на настоящее время задачи, по подходам к ее решению. К сожалению, ограниченный объем публикации и ряд других объективных причин не позволяют представить модели, методы и средства, составляющие такие подходы, более детально. Одна из объективных причин — отсутствие целостного взгляда на подобные методы и средства среди исследователей, которые занимаются или должны заниматься этими вопросами и в России, и за рубежом. Пока идет процесс накопления знаний, которые необходимы для точной (строгой) постановки задачи и выбора адекватных подходов к ее решению.

С одной стороны, сложившееся положение дел объясняется глобальным характером, транснациональной значимостью задачи, а также существенно междисциплинарным характером вопросов, которые стоят на пути ее решения. С другой стороны, и это отмечено ранее, уже есть реальные результаты исследований аналогичного характера по тематике кибертерроризма. Они систематизированы и достаточно подробно отражены в открытых публикациях и представлены в Интернет. Результаты российских исследователей, выполненные с участием автора, изложены, например, в двухтомном издании [11, 12]. С этих позиций одна из целей настоящей работы — привлечь внимание исследователей, имеющих знания и опыт работы в области традиционной безопасности информационных технологий, к этой задаче, включиться в ее решение. Второй целью является призыв к структурам органов государственного управления, которых эта задача касается напрямую, включить ее в перечень приоритетных задач для исследования. В противном случае, как это происходило неоднократно в нашей стране, придется догонять "уже ушедший поезд".

В связи с изложенным выше можно сделать следующие выводы.

1. Проведение консультаций, исследование вопросов, направленных на создание эффективной, международно признанной системы противодействия крупномасштабным киберконфликтам террористического или военно-политического характера в информационном пространстве, должно осуществляться с позиций предварительного системного представления этой проблемной области. Сложившиеся на настоящее время подходы (механизмы и модели, методы и средства) к проведению консультаций и выработке приемлемых на транснациональном уровне решений этим требованиям пока не удовлетворяют, носят фрагментарный характер. Это обстоятельство затрудняет получение конечного результата.

2. Для эффективной систематизации и более строгого описания этой области необходимо проводить скоординированные исследования группами специалистов, владеющих методами информатики, политологии и государственного управления.

3. В качестве базовых сущностей, определяющих проблемную область, в первоочередном порядке целесообразно рассматривать: критически важное киберпространство, критически важные инфраструктуры и составляющие ее объекты; модели киберугроз этим объектам; их источники и сценарии реализации. Такие модели должны включать пороговые значения, отображающие многоплановые ущербы от реализации угроз, превышение которых констатирует факт наличия крупномасштабного киберконфликта и дает легитимные основания для принятия ответных киберконтрмер.

4. Программные средства, которые призваны реализовать предлагаемые в настоящей работе модели и средства защиты от деструктивных кибервоздействий в военно-политических целях, должны удовлетворять современным стандартам программной инженерии. Они должны быть не только отчуждаемы, масштабируемы и верифицируемы, но и легко модифицируемы и адаптируемы к тем требованиям, в рамках которых создаются программные продукты аналогичного назначения в других странах. Как следствие, такие продукты должны иметь хорошие перспективы интеграции.

Работа выполнена при финансовой поддержке государственного контракта 07.514.11.4146 Министерства образования и науки РФ.

Список литературы

1. **Материалы** первой—шестой Международных конференций по проблемам безопасности и противодействия терроризму. Московский государственный университет им. М. В. Ломоносова. М.: МЦНМО, 2005—2010 гг.

2. **Security**, Terrorism and Privacy in Information Society ISC 2005 // Proc. the Third International Security Conference, Dussaldorf 27—28 October 2005. W.Bertelsmann Verlag GmbH Co, KG, Bielefeld, 2006. 486 p.

3. **Газета "Московский университет"**. 2011 июнь. № 20 (4371).

4. **Mallery J. C.** Possible International Workshops On Critical Cyber Policy Issues // Fourth International Forum "Partnership between State Authorities, Civil Society, and Business Community in Ensuring Information Security and Combating Terrorism" Garmisch-Partenkirchen, Germany, April 12—15, 2010. URL: http://www.iisi.msu.ru/UserFiles/File/bayern2010/presentations/mallery.ppt

5. **Mallery J. C.** Straw Man Architecture For An International Cyber Data Sharing System // INCO-Trust "Workshop On International Cooperation In Security And Privacy.: International Data Exchange With Security And Privacy: Applications, Policy, Technology, and Use," New York Academy of Sciences, New York City, May 3—5, 2010. URL: http://www.cs.rutgers.edu/~rwright1/INCO-TRUST/Position/Mallery.ppt

6. **Mallery J. C.** A Strategy for Cyber Defense Strategy // 2010 Workshop on Cyber Security and Global Affairs & Security Confabulation IV, Zurich, July 7—9, 2010. URL: http://icc.ite.gmu.edu/csga2010/John_Mallery.ppt

7. **Mallery J. C.** Dialectics of Cyber International Relations and Cyber Defense: Towards a Strategic Research Program // ECIR Workshop on Cyber International Relations: Emergent Realities of Conflict and Cooperation, MIT, Cambridge, October 14, 2010. URL: http://web.mit.edu/ecir/pdf/ecir-mallery5.pptx

8. **Mallery J. C.** Trustworthy Cloud Computing: Risks, Challenges and Recommendations // 2011 Workshop on Cyber Security and Global Affairs, Budapest, Hungary, May 31 — June 2, 2011. URL: http://icc.ite.gmu.edu/csga2011/Cloud-John_Mallery.pdf

9. **Mallery J. C., Janssen T.** Norms in the United States International Strategy for Cyberspace // 2011 Workshop on Cyber Security and Global Affairs, Budapest, Hungary, May 31 — June 2, 2011. URL: http://icc.ite.gmu.edu/csga2011/Mallery_Janssen.pdf

10. **Mallery J. C.** Models of Escalation and De-escalation in Cyber Conflict // 2011 Workshop on Cyber Security and Global Affairs, Budapest, Hungary, May 31 — June 2, 2011. URL: http://icc.ite.gmu.edu/csga2011/Cyber_Conflict_John_Mallery.pptx

11. **Критически** важные объекты и кибертерроризм. Часть 1. Системный подход к организации противодействия / О. О. Андреев. И. С. Батов, М. В. Большаков и др. Под ред. В. А. Васенина. М.: МЦНМ. 2008. 398 с.

12. **Критически** важные объекты и кибертерроризм. Часть 2. Аспекты программной реализации средств противодействия / О. О. Андреев, М. С. Астапов, А. С. Афонин и др. Под ред. В. А. Васенина. М.: МЦНМ. 2008. — 607 с.

13. **ГОСТ Р ИСО/МЭК 15408-1—2002.** Информационная технология. Методы и средства обеспечения безопасности. Критерии оценки безопасности информационных технологий. Часть 1. Введение и общая модель. М.: ИПК Издательство стандартов, 2002.

14. **ГОСТ Р ИСО/МЭК 15408-2—2002.** Информационная технология. Методы и средства обеспечения безопасности. Критерии оценки безопасности информационных технологий. Часть 2. Функциональные требования безопасности. — М.: ИПК Издательство стандартов, 2002.

15. **ГОСТ Р ИСО/МЭК 15408-3—2002.** Информационная технология. Методы и средства обеспечения безопасности. Критерии оценки безопасности информационных технологий. Часть 3. Требования доверия к безопасности. — М.: ИПК Издательство стандартов, 2002.

16. **ГОСТ Р ИСО/МЭК 13335-1—2006.** Информационная технология. Методы и средства обеспечения безопасности. Часть 1. Концепция и модели менеджмента безопасности информационных и телекоммуникационных технологий. — М.: ИПК Издательство стандартов, 2006.

17. **ГОСТ Р ИСО/МЭК 17799—2005.** Информационная технология. Практические правила управления информационной безопасностью. — М.: ИПК Издательство стандартов, 2005.

18. **Средства** вычислительной техники. Защита от несанкционированного доступа к информации. Показатели защищенности от несанкционированного доступа к информации. Руководящий документ ФСТЭК от 30 марта 1992 года / ФСТЭК. — 1992.

19. **Автоматизированные** системы. Защита от несанкционированного доступа к информации. Классификация автоматизированных систем и требования по защите информации. Руководящий документ ФСТЭК от 30 марта 1992 года / ФСТЭК. — 1992.

20. **The National** Strategy to Secure Cyberspace. — White House, Washington, February 2003.

21. **Крутских А. В., Алексеева И. Ю.** Информационные вызовы национальной и международной безопасности. Под общ. ред. А. В. Федорова, В. Н. Цыгичко. М.: ПИР-Центр, 2001. 328 с.

22. **Крутских А. В., Зиновьева Е. С.** Политические проблемы развития мировой научно-технологической сферы // Современные глобальные проблемы / Отв. ред. В. Г. Барановский, А. Д. Богатуров, ред. А. С. Дундич. М.: Аспект-Пресс, 2010.

23. **Грушо А. А., Тимонина Е. Е.** Теоретические основы защиты информации. М.: Изд-во "Яхтсмен", 1996, 192 с.

24. **Девянин П. Н.** Анализ безопасности управления доступом и информационными потоками в компьютерных системах. М.: Радио и связь, 2006. 176 с.

25. **Шапченко К. А.** Способ проверки свойств безопасности в моделях логического разграничения доступа с древовидной иерархией объектов доступа // Информационные технологии. 2009. № 10. С. 13—17.

26. **Васенин В. А.** Модернизация экономики и новые аспекты инженерии программ // Программная инженерия. 2012. № 2. С. 2—18.

27. http://www.ewi.info/cybersecurity-terminology-foundations

28. **Васенин В. А., Галатенко А. В.** Компьютерный терроризм и проблемы информационной безопасности в Интернет / Высокотехнологичный терроризм. Материалы российско-американского семинара. Москва, 4—6 июня 2001. РАН в сотр. с нац. Академиями США, 2001. С. 211—225.

29. **Васенин В. А.** Проблемы безопасности информационных технологий: анализ "узких" мест // Комитет Совета Федерации по обороне и безопасности. Научные основы национальной безопасности Российской Федерации. Материалы семинара, Москва 24 мая 2005 г. М., 2005. С. 70—74.

УДК 004:351

Я. Н. Имамвердиев, канд. техн. наук, зав. отделом,
Институт информационных технологий НАНА,
г. Баку,
e-mail: yadigar@lan.ab.az

Метод обнаружения переделанных отпечатков пальцев на основе фрактальных характеристик

Рассмотрена возможность использования концепций теории фракталов для описания свойств отпечатков пальцев. Разработан метод определения фрактальной размерности отпечатков пальцев и на его основе предложен эффективный подход для обнаружения переделанных отпечатков пальцев. Результаты экспериментов показывают, что метод хорошо отличает изображения реальных отпечатков пальцев от переделанных. Предложенный метод не требует дополнительного оборудования и легко встраивается в существующие системы распознавания отпечатков пальцев.

***Ключевые слова**: переделанный отпечаток пальца, фрактал, фрактальная размерность, мультифрактальный спектр, обнаружение переделанных отпечатков пальцев, машина опорных векторов*

Введение

В связи с широким применением биометрических технологий вопросы их безопасности становятся более актуальными. Хотя в последние годы различные вопросы безопасности биометрических технологий исследовались достаточно широко, включая методы обнаружения фальшивых биометрических образцов и методы защиты биометрических данных, некоторые вопросы остались без должного внимания, например, вопросы обнаружения переделанных отпечатков пальцев рассматривались в научной литературе впервые только в 2009 г. [1]. Переделка отпечатков пальцев означает умышленное изменение узоров отпечатков пальцев в целях маскировки личности. Использование переделанных отпечатков пальцев имеют давнюю историю [2] и неоднократно встречались на практике миграционных и правоохранительных органов, о нескольких таких случаях сообщалось в прессе [1].

Следует отметить, что переделанные отпечатки пальцев отличаются от фальшивых (муляжей) отпечатков пальцев. Фальшивые отпечатки пальцев, изготовленные из желатина, латекса, силикона и т. д., обычно используются в целях выдачи себя за другую личность. Переделанные отпечатки являются реальными отпечатками и используются для маскировки собственной личности в целях уклонения от идентификации со стороны биометрической системы.

Для обнаружения фальшивых отпечатков пальцев существуют различные методы на основе программного или аппаратного обеспечения [3], в работе [4] дан обзор этих методов. Методы обнаружения переделанных отпечатков пальцев пока широко не изучались [1, 5—7]. Программное обеспечение по оценке качества изображения отпечатков пальцев (например, NFIQ [8]) не всегда может обнаружить переделанные отпечатки, так как качество изображений отпечатков может не измениться в результате переделки.

Попытки использования переделанных отпечатков пальцев входят в более широкую категорию атак, известную как *биометрическая обфускация* [5]. Биометрическую обфускацию можно определить как умышленную попытку индивидуумов маскировать свою личность от биометрической системы путем переделки своей биометрической характеристики. Примерами биометрической обфускации являются изменения узоров отпечатков пальцев, нарушение радужной оболочки глаза с помощью театральных линз, изменение лица с помощью пластических операций. Было сообщено, что пластические операции могут значительно деградировать производительность систем распознавания лиц [9], а операции при катаракте могут уменьшить точность систем распознавания радужной оболочки глаза [10].

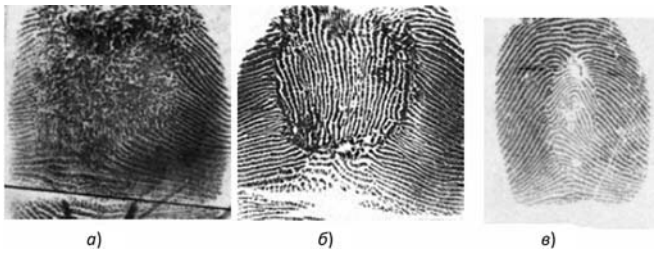
Распознавание по отпечаткам пальцев является самой распространенной биометрической технологией. Системы автоматического распознавания отпечатков пальцев (Automated Fingerprint Identification System, AFIS) занимают почти половину международного рынка биометрических систем [11]. Эти системы базируются на предположениях, что отпечатки пальцев уникальны и не изменяются на протяжении всей жизни взрослого человека. Использование переделанных отпечатков пальцев подрывает эти предположения и представляет угрозу для надежности и безопасности AFIS. Кроме того, в сравнении с радужной оболочкой глаза или лица, где требуются хирургические операции, отпечатки пальцев относительно легко переделать, используя, например, абразивные или химические материалы.

Целью настоящей статьи является разработка метода обнаружения переделанных отпечатков пальцев на основе фрактальных характеристик. В результате переделки отпечатков пальцев образуются новые структурные элементы: шрамы, области разрушения папиллярных узоров, зоны разрывного изменения поля ориентаций отпечатков пальцев и т. д. Гипотеза исследования основывается на этом наблюдении и исходит из способности фрактального анализа [12] детально описывать локальную и глобальную пространственную структуру сложных систем.

Типы переделанных отпечатков пальцев

В работе [5] переделанные отпечатки пальцев разделены на три класса на основе изменений гребневого узора в результате переделки:

- стертые отпечатки пальцев;
- искаженные отпечатки пальцев;
- имитированные отпечатки пальцев.



Образцы переделанных отпечатков пальцев [5]:
а — искажение; *б* — трансплантация кожи из ладони; *в* — изысканная трансплантация из другой гребневой кожи

Самым популярным методом переделки отпечатков пальцев является стирание узоров отпечатков с помощью трения, выжигания, резания, применения химикатов и трансплантации гладкой кожи (см. рисунок). Кожные болезни (проказа) и побочные эффекты лекарств против рака могут также стирать отпечатки пальцев. Гребневая структура узоров едва видна в стертой области.

Искаженные отпечатки пальцев получают удалением части кожи на пальце и заменой ее кожей из ладони или подошвы или пересадкой ее в другом положении. Искаженные отпечатки пальцев имеют необычные образы гребней, которые не встречаются в реальных отпечатках пальцев. Эти особенности включают анормальное пространственное распределение сингулярных точек (точки дельта и ядро) или резкие изменения поля ориентаций вдоль шрамов. Заметим, что разрывы поля ориентаций в естественных отпечатках наблюдаются только в сингулярных точках.

Гребневой узор может сохранить свою похожесть на узор отпечатков пальцев после искусно сделанной процедуры переделки отпечатков. Например, часть кожи удаляется, остальные части натягиваются и зашиваются вместе, или трансплантируется весь отпечаток пальца. Гребневая кожа из других частей используется для заполнения удаленной кожи на пальце с сохранением согласованности узоров. Например, в работе [5] сообщается, что просто обменом кожи пальцев на правой и левой руках удалось обмануть AFIS.

Имитированные отпечатки пальцев не только успешно проходят проверки программного обеспечения по оценке качества изображения, они могут спутать даже экспертов по отпечаткам пальцев.

Переделанный отпечаток пальца на рис. 1, *в* имеет очень гладкое поле ориентаций и единственным свидетельством возможной переделки является тонкий шрам. Этот отпечаток получился в результате изысканной хирургической процедуры, он имеет естественное течение гребней даже вдоль хирургических шрамов.

Обзор работ по обнаружению переделанных отпечатков пальцев

Как уже отмечалось, имеется всего несколько работ по обнаружению переделанных отпечатков пальцев. Предложенный в статьях [5, 6] метод автоматического обнаружения переделанных отпечатков использует два признака.

Первый признак основан на анализе поля ориентаций отпечатков пальцев. Отпечатки пальцев хорошего качества имеют гладкое поле ориентаций, кроме окрестностей сингулярных точек. На основе этого факта были разработаны многочисленные модели поля ориентаций отпечатков пальцев [13—15], которые комбинируют модель глобального поля ориентаций для непрерывного поля и локального поля ориентаций в окрестностях сингулярных точек. Если для аппроксимации поля ориентаций использовать только глобальную модель, то разность между наблюдаемым и вычисленным по глобальной модели полем ориентаций будет отлична от нуля только вокруг сингулярных точек.

Для переделанных отпечатков пальцев ошибка подгонки модели также наблюдается в переделанных областях, поэтому разность между полем ориентаций, извлеченным из изображения пальца, и полем, аппроксимированным глобальной моделью, можно использовать как вектор признаков для классификации отпечатков пальцев как оригинальный или переделанный.

Второй вектор признаков основан на наблюдении, что распределение минуций в переделанных отпечатках пальцев часто отличается от распределения на естественных пальцах.

В работе [7] для обнаружения переделанных отпечатков пальцев предлагается метод на основе надежности поля ориентаций. Карта надежности поля ориентаций отпечатков пальцев имеет пики в сингулярных точках. Эти пики используются для анализа переделанных пальцев, так как в результате переделки появляются другие сингулярные точки — пики с малой амплитудой.

Предложенные в работах [5, 6] методы основываются на нахождении сингулярных точек, но эта задача сама является достаточно сложной, а отпечатки пальцев дугового типа не имеют сингулярных точек. Кроме того, в отпечатках пальцев, полученных от сканеров, могут отсутствовать точки дельта из-за малой площади сенсора. Метод, предложенный в работе [7], также опирается на сингулярные точки.

Фракталы и методы оценки фрактальной размерности

Теория фракталов и ее приложения к различным объектам в науке, технике и в технологиях получили в последние годы широкое распространение [16—20]. Понятие о фракталах (лат. fractus — дробленый, сломанный, разбитый) ввел Б. Мандельброт в 1960-1970-е годы, с выходом его книги "The Fractal Geometry of Nature" [16] принято связывать рождение фрактальной геометрии. Мандельброту удалось объединить в единую систему научные результаты других ученых, работавших в той же области ранее (Пуанкаре, Фату, Жюлиа, Кантор, Хаусдорф).

Сам Мандельброт давал в своих работах разные определения фракталам, определяя их как объекты, которые [16, 17]:

- имеют очень фрагментарную или нерегулярную форму;
- обладают самоподобием или самоаффинностью;
- обладают масштабной инвариантностью.

Масштабная инвариантность предполагает неизменность основных геометрических особенностей фракталов при масштабных переходах. Самоподобие является простейшим примером масштабной инвариантности. Самоподобие означает, что структура фрактала в одном масштабе подобна его структуре в другом масштабе. Отметим, что это свойство приводит к степенным зависимостям. Самоаффинность является обобщением самоподобия и является более сложной формой масштабной инвариантности. Часть афинно-самоподобного объекта подобна целому объекту после аффинного преобразования.

Существует множество классификаций фракталов, их можно разделить на два основных класса: регулярные (детерминистические) и нерегулярные (случайные) фракталы. Регулярные фракталы являются математическими абстракциями, они точно конструируются на основе некоторых базовых геометрических или алгебраических преобразований. Например, для построения фрактальных структур широко используется метод "систем итерируемых функций" (Iterated Functions System, IFS). Фракталы IFS строятся на основе простых преобразований плоскости: масштабирования, перемещения и вращения осей плоскости. Типичными примерами регулярных фракталов являются множество Кантора, кривая Коха, ковер Серпинского и др. Наиболее важными свойствами регулярных фракталов является возможность точного расчета их фрактальной размерности и свойство точного самоподобия в любых масштабах.

Нерегулярные (случайные) фракталы в отличие от регулярных обладают способностью к самоподобию в ограниченных пределах, определяемых реальными размерами системы. При этом увеличенная часть фрактала не точно идентична исходному фрагменту, однако их статистические характеристики совпадают. Случайные фракталы можно получить, если вместо детерминированного способа построения фракталов включить в алгоритм их создания некоторый элемент случайности.

Основным параметром, характеризующим фрактал, является фрактальная размерность, описывающая сохраняемость статистических характеристик при изменении масштаба. В качестве фрактальной размерности используется размерность, введенная Ф. Хаусдорфом еще в 1918 г. для компактных множеств в метрических пространствах:

$$D = \lim_{\delta \to 0} \frac{N(\delta)}{1/\delta}, \qquad (1)$$

где $N(\delta)$ — минимальное число шаров радиуса δ, покрывающих изучаемое множество. В евклидовых пространствах помимо шаров для покрытия можно использовать и другие элементы с характерными линейными размерами δ.

В настоящее время для вычисления значений фрактальной размерности используют множество методов, в том числе метод считывания блоков (box-counting method, BC-метод), метод Херста (R/S method — метод нормированного размаха), Фурье-анализ профилей, метод вертикальных сечений и т. д.

BC-метод (в русскоязычной литературе также используется название "метод покрытия квадратами") используется наиболее часто. Данный метод покрывает множество d-мерными блоками с длиной ребра δ, подсчитывает их числа $N(\delta)$, строит график в логарифмических координатах $\log(N(\delta))$, $\log(\delta)$ и определяет по углу наклона фрактальную размерность:

$$D = -\Delta\log(N(\delta))/\log(\delta). \qquad (2)$$

Существуют разные варианты BC-метода, они различаются параметрами (например, в быстром алгоритме считывания блоков длина стороны блока варьируется как 2^k, $1 \leqslant k \leqslant K$, где 2^k — размер изображения) и моделями изображения (например, без интерполяции между точками данных и т. д.).

Многие естественные текстурные поверхности можно описывать как фрактальные поверхности. Фрактальная размерность является полезным признаком для сегментации текстур, классификации форм и графического анализа во многих областях. Как показало исследование в [21], фрактальная размерность очень хорошо коррелирует с оценкой человеком "неровности" поверхности.

В работе [22] была исследована возможность использования фрактальной размерности для анализа отпечатков пальцев. В [23] используется не одиночная оценка фрактальной размерности, а вектор признаков на основе фрактальной размерности для биометрической идентификации. Высказывается предположение, что мультифракталы являются более подходящими для описания сложных текстурных изображений, в том числе изображений отпечатков пальцев [12, 24].

Мультифракталы являются обобщением фракталов, они отличаются от обычных фракталов наличием зависимости геометрических свойств от масштабного уровня. Эта зависимость может быть детерминированной или стохастической и приводит к тому, что метрические свойства мультифрактала характеризуются не одиночной фрактальной размерностью, а спектром фрактальных размерностей (Multifractal spectrum, MFS). Фактически, мультифрактальный подход означает, что изучаемый объект каким-то образом можно разделить на вложенные фрагменты (подмножества), для каждого из которых наблюдаются свои свойства самоподобия [12].

Метод обнаружения переделанных отпечатков пальцев на основе фрактальных характеристик

В этом разделе описывается метод обнаружения переделанных отпечатков пальцев на основе фрак-

тальных характеристик. Предлагаемый метод можно представить в виде последовательности шагов.

1. Входное изображение отпечатка пальца I предварительно обрабатывается и нормализуется с применением методов, предложенных в работе [25].

2. Вычисляется фрактальная характеристика, и на ее основе формируется вектор признаков.

3. Вектор признаков классифицируется с помощью SVM-классификатора.

В качестве фрактальных характеристик для обнаружения переделанных отпечатков пальцев предлагается использовать вектор признаков на основе:

- фрактальной размерности, вычисленной модифицированным методом Катца;
- локальной экспоненты масштабирования;
- мультифрактального спектра.

Вычисление вектора признаков на основе фрактальной размерности. В изображении отпечатков пальцев гребни и впадины в локальном соседстве формируют плоскую синусоидальную волну. Для вычисления фрактальной размерности волнообразных сигналов предложены различные алгоритмы (Katz, Higuchi, Sevcik и др.) [27—28], в [29] проводится их сравнительный анализ. В данной работе используется вариант метода Katz для изображений отпечатков пальцев [23].

Пусть изображение отпечатков пальцев размера $N \times N$ бинаризовано (размер изображения 256×256, разрешение 500 dpi) и разбито на N строк и N столбцов. Можно считать, что в каждой строке встречается бинарное значение 1 (в бинаризованном изображении пиксели гребня задаются значением 1, а пиксели впадин — значением 0).

Для каждой строки вычисляется фрактальная размерность по следующей формуле:

$$d_n = \frac{\log_{10}(N)}{\log_{10}(M_n)},$$

где M_n — общее число бинарных значений 1 в строке n.

Вектор признаков для всего изображения отпечатков пальцев можно построить как

$$\Phi = \bigcup_{n=1}^{N} (n', d_n);$$

$$n' = \left\lfloor \frac{(d_n - 1)}{(n+1) - n} \right\rfloor n = (d_n - 1)n,$$

где n' — горизонтальный коэффициент масштабирования, $0 < n' < N$.

Вычисление вектора признаков на основе локальных экспонент Гёльдера. Некоторые исследования показали, что одиночная фрактальная размерность не захватывает все текстурные свойства изображений [24]. Можно ожидать, что из-за значительных изменений структуры переделанных отпечатков пальцев фрактальная размерность, вычисленная методом считывания блоков, будет изменяться в зависимости от размера блока. Если множество имеет некоторое фрактальное свойство при ограниченном интервале масштабирования, то это свойство оценивают с помощью локальной экспоненты масштабирования. Отметим, что локальную экспоненту масштабирования часто называют локальной экспонентой Гёльдера, она характеризует локальную регулярность изображения [12].

Для вычисления локальных экспонент Гёльдера существуют различные методы, в этой работе используется BC-метод. В случае чистой фрактальной размерности D число блоков $N(\delta)$, необходимых для покрытия, следует степенному закону $N(\delta) = \delta^{-D}$, а в реальных системах это отношение сохраняется только в ограниченном интервале масштабов. Если определить локальную экспоненту масштабирования как логарифмический наклон

$$D(\delta) = -\frac{\ln N(\delta)}{\ln \delta},$$

то интервалом фрактальности будет множество масштабов, где этот наклон является приблизительно постоянным.

Вычисление вектора признаков на основе мультифрактального спектра. Фрактальное свойство можно описывать более детально мультифрактальным спектром (спектром сингулярностей) или распределением вероятностей локальных экспонент Гёльдера. Для вычисления мультифрактальных спектров часто используют вейвлет-преобразования. Известно, что все вейвлеты данного семейства $\psi_{j,\,k}(x)$ подобны своему базисному вейвлету $\psi(x)$ и получаются из него с помощью сжатий и сдвигов. Вейвлет-анализ изучает поведение сигналов на разных масштабах путем вычисления скалярного произведения, анализирующего вейвлета на исследуемый сигнал, поэтому он очень хорошо подходит для исследования фрактального поведения. В терминах вейвлет-коэффициентов это подразумевает степенное поведение их высших моментов при изменении масштаба [30].

В этой работе используются двумерные дискретные вейвлет-преобразования (Discrete Wavelet Transform, DWT) [31]. DWT декомпозирует входное изображение I в одну низкочастотную компоненту $D_J(I)$ и несколько высокочастотных компонент под несколькими масштабами: $W_{k,\,j}(I)$, $k = 1, 2, 3$; $j = 1, 2, \ldots, J$, где J — число масштабов (в этой работе используется $J = 3$). Таким образом, имеем три высокочастотных компоненты ($k = 1, 2, 3$) на каждом уровне масштаба, которые описывают разрывы изображения вдоль горизонтального, вертикального и диагонального направлений. При реализации предложенного метода были использованы вейвлеты Добеши ‘DB2’.

Для вычисления мультифрактальных спектров предлагается использовать следующую пирамиду вейвлетов [32, 33]: низкочастотные вейвлеты $D(I)$, высокочастотные вейвлеты $W(I)$ и вейвлет-лидеры $L(I)$:

$$\{D_J,\ W_{k,\,j},\ L_j\}. \qquad (3)$$

Для каждой компоненты пирамиды вейвлетов вычисляется MFS-спектр ВС-методом:

$$\{MFS(D),\ MFS(W_k),\ MFS(L)\}. \qquad (4)$$

Вейвлет-лидер определяется как максимальный отклик всех вейвлет-коэффициентов в пространственном и скейлинг-соседствах в более малых масштабах. Иными словами, для вейвлет-коэффициента $W_{k,j_0}(r_0)$ в пикселе r_0 соответстующий вейвлет-лидер определяется следующим образом:

$$L_{j_0}(r_0) = \max_{1 \leqslant j \leqslant j_0 \leqslant 1} \max_{k \leqslant 3} \max_{r \in \Omega(r_0)} |W_{k,j}(r_0)|, \qquad (5)$$

где $\Omega(r_0)$ — квадратное соседство с центром в r_0.

Для каждой матрицы вейвлет-коэффициентов при использовании алгоритма, описанного в [33], получается 26-мерный MFS-спектр. Так как используются три масштаба ($J = 3$) и имеются 13 матриц вейвлет-коэффициентов, включая низкочастотные, высокочастотные компоненты и вейвлет-лидеры, в результате получается $26 \times 13 = 338$-мерный вектор признаков. В этой работе используется алгоритм выбора признаков для SVM [34], при этом получается 103-мерный вектор признаков для представления отпечатков пальцев.

Вычислительные эксперименты

Отметим, что отсутствие доступных открытых баз данных по переделанным отпечаткам пальцев ставит в трудное положение исследователей в этой области. Эксперименты в работах [1] были проведены на базе данных синтетически созданных переделанных отпечатков пальцев. В работе [5] использована база данных по переделанным отпечаткам пальцев, но пока эта база данных не доступна для широкого круга исследователей.

Для экспериментальной проверки предложенного метода был создан набор синтетических отпечатков пальцев на основе изображений отпечатков пальцев размером 640×480 из базы данных FVC2002-DB1, полученных на оптическом сканере при разрешении 500 dpi.

Были имитированы два типа переделки [1, 7]:

- Z-вырезка (получается вырезанием в форме Z, пересоединением двух треугольников и зашиванием их обратно);
- центральное вращение (получается вырезанием круглой области в центре изображения и ее вращением).

Предложенный метод был реализован на MATLAB R2008b, использовался процессор Intel ® Core (TM)2 Quad CPU Q6600 @ 2.4 GHz с оперативной памятью 2 GB RAM. Все шаги алгоритма выполнялись примерно за 3 с.

Для вычисления мультифрактальных спектров были использованы свободно распространяемое программное обеспечение Fraclab [35] и MATLAB-коды для ВС-метода [36].

Для SVM-классификации с 10-кратной кросс-валидацией была использована программа LIBSVM [37] с радиальной базисной ядерной функцией. Результаты LIBSVM были линейно масштабированы к интервалу [0, 1]. Когда нормализованное значение для входного отпечатка пальца меньше предопределенного порогового значения, то выдается сигнал о том, что изображение, возможно, является переделанным отпечатком.

Частота верных положительных классификаций переделанных отпечатков пальцев

Метод вычисления вектора признаков	Частота ложных положительных классификаций						
	0,1	0,2	0,3	0,5	1	2	3
Фрактальная размерность (метод Каца)	38	42	44	50	60	76	79
Локальные экспоненты Гёльдера	45	57	60	64	75	85	88
Мультифрактальный спектр	54	65	68	73	81	90	93

Результаты вычислительных экспериментов для разных значений представлены в таблице. Для оценки качества классификации в таблице приведены значения частоты верных положительных классификаций (True Positive Rate) при заданной частоте ложных положительных классификаций (False Positive Rate) [38, 39].

Заключение

Предложен метод для обнаружения переделанных отпечатков пальцев на основе фрактальных характеристик. Впервые опробован мультифрактальный анализ для количественной параметризации отпечатков пальцев и установлено, что использование мультифрактальных спектров эффективно отражает изменения структуры отпечатков пальцев, происходящие в результате переделки. Результаты экспериментов подтверждают эффективность предложенного метода, который можно достаточно легко встраивать в существующие системы распознавания отпечатков пальцев, не ухудшая их производительности.

Список литературы

1. **Feng J., Jain A. K., Ross A.** Fingerprint alteration // MSU-TechnicalReport. MSU-CSE-09-30. Dec. 2009.
2. **Cummins H.** Attempts to Alter and Obliterate Fingerprints // Journal of American Institute of Criminal Law and Criminology. 1935. V. 25. P. 982—991.
3. **Imamverdiyev Y. N., Kerimova L. E., Musayev V. Y.** Method of detection of real fingerprints on the basis of the Radon transform // Automatic Control and Computer Sciences. 2009. V. 43. N 5. P. 270—275.
4. **Алгулиев Р. М., Имамвердиев Я. Н., Мусаев В. Я.** Методы обнаружения живучести в биометрических системах // Вопросы защиты информации, 2009. № 3 (86). С. 16—21.
5. **Yoon S., Feng J., Jain A. K.** Altered fingerprints: analysis and detection // IEEE Transactions on Pattern Analysis and Machine Intelligence. 2011.
6. **Feng J., Jain A. K., Ross A.** Detecting Altered Fingerprints // Proc. 20th International Conference on Pattern Recognition (ICPR). August 23—26, Istanbul, Turkey. 2010. P. 1622—1625.
7. **Petrovici A., Lazar C.** Identifying Fingerprint Alteration Using the Reliability Map of the Orientation Field // The Annals of the Univeristy of Craiova, Series Automation, Computers, Electronics and Mechatronics. 2010. V. 7 (34). N 1. P. 45—52.

8. **Tabassi E., Wilson C., Watson C.** Fingerprint Image Quality, NISTIR 7151, August 2004. URL: http://fingerprint.nist.gov/NFIS/ir 7151.pdf.
9. **Singh R., Vatsa M., Bhatt H. S., Bharadwaj S., Noore A., Nooreyezdan S. S.** Plastic Surgery: A New Dimension to Face Recognition // IEEE Trans. Information Forensics and Security. 2010. V. 5. N 3. P. 441—448.
10. **Roizenblatt R., Schor P., Dante F., Roizenblatt J., Belfort R.** Iris Recognition As a Biometric Method after Cataract Surgery // American Journal of Ophthalmology. 2005. V. 140. N 5. P. 969.
11. **Maltoni D., Maio D., Jain A. K., Prabhakar S.** Handbook of Fingerprint Recognition (Second Edition). Springer-Verlag. 2009.
12. **Павлов А. Н., Анищенко В. С.,** Мультифрактальный анализ сложных сигналов // Успехи физических наук. 2007. Т. 177. № 8. С. 859—876.
13. **Bazen A. M., Gerez S. H.** Systematic Methods for the Computation of the Directional Fields and Singular Points of Fingerprints // IEEE Trans. Pattern Analysis and Machine Intelligence. 2002. V. 24. N 7. P. 905—919.
14. **Zhou J., Gu J.** A Model-Based Method for the Computation of Fingerprints' Orientation Field // IEEE Trans. Image Processing. 2004. V. 13. N 6. P. 821—835.
15. **Wang Y., Hu J.** Global Ridge Orientation Modeling for Partial Fingerprint Identification // IEEE Trans. Pattern Analysis and Machine Intelligence. 2010. V. 33. N 1. P. 72—87.
16. **Mandelbrot B. B.** The Fractal Geometry of Nature. San-Francisco: W. H. Freeman and Comp. 1982. 459 p.
17. **Mandelbrot B. B.** A Multifractal Walk down Wall Street // Scientific American. Feb. 1999. P. 70—73.
18. **Кроновер Д. М.** Фракталы и хаос в динамических системах. Основы теории. М.: Постмаркет. 2000. 352 с.
19. **Федер Е.** Фракталы / Пер. с англ. М.: Мир. 1991. 260 с.
20. **Петерс Э.** Фрактальный анализ финансовых рынков: применение теории хаоса в инвестициях и экономике. М.: Интернет-трейдинг. 2004. 304 с.
21. **Pentland A.** Fractal-Based Description of Natural Scenes // IEEE Transactions on Pattern Analysis and Machine Recognition. 1984. V. 6. N 6. P. 661—674.
22. **Polikarpova N.** On the Fractal Features in Fingerprint Analysis // Proc. of the 13th Int. Conf. on Pattern Recognition. 1996. V. 3. P. 591—595.
23. **Lin C.-H., Chen J.-L., Gaing Z.-L.** Combining Biometric Fractal Pattern and Particle Swarm Optimization-Based Classier for Fingerprint Recognition // Hindawi Publishing Corporation. Mathematical Problems in Engineering. 2010. V. 2010. Article ID 328676. 14 p.
24. **Lopes R., Betrouni N.** Fractal and multifractal analysis: A review // Medical Image Analysis. 2009. V. 13. P. 634—649.
25. **Hong L., Wan Y., Jain A.** Fingerprint image enhancement: algorithm and performance evaluation // IEEE Trans Pattern Anal Mach Intelligence. 1998. V. 20. N 8. P. 777—789.
26. **Katz M. J.** Fractals and the analysis of waveforms // Computers in Biology and Medicine. 1988. V. 18. N 3. P. 145—156.
27. **Higuchi T.** Approach to an irregular time series on the basis of the fractal theory // Physica D: Nonlinear Phenomena. 1988. V. 31. N 2. P. 277—283.
28. **Sevcik C.** On fractal dimension of waveforms // Chaos, Solitons and Fractals, 2006. V. 28. N 2. P. 579—580.
29. **Esteller R., Vachtsevanos G., Echauz J., Litt B.** A Comparison of Waveform Fractal Dimension Algorithms // IEEE Trans. Circuits Syst. — I: Fundam. Theory Appl. 2001. V. 48. N 2. P. 177—183.
30. **Struzik Z. R.** Determining local singularity strengths and their spectra with the wavelet transform // Fractals. 2000. V. 82. P. 163—179.
31. **Дремин И. М., Иванов О. В., Нечитайло В. А.** Вейвлеты и их использование // Успехи физических наук. 2001. Т. 171. N 5. С. 465—501.
32. **Xu Y., Ji H., Fermuller C.** Viewpoint Invariant Texture Description Using Fractal Analysis // Int. Journal of Computer Vision. 2009. V. 83. N 1. P. 85—100.
33. **Wendt H., Abry P., Jaffard S., Ji H., Shen Z.** Wavelet Leader Multifractal Analysis for Texture Classification // Proc. of the 16th IEEE Int. Conference on Image Processing (ICIP). 2009. P. 3785—3788.
34. **Chen Y. W., Lin C. J.** Combining SVMs with various feature selection strategies // Studies in Fuzziness and Soft Computing. 2006. V. 207. P. 315—324.
35. **FracLab 2.0.** A fractal analysis toolbox for signal and image processing. URL: http://fraclab.saclay.inria.fr/
36. **Moisy F.** Computing a fractal dimension with Matlab: 1D, 2D and 3D Box-counting. URL: http://www.mathworks.com/matlabcentral/fileexchange/13063-boxcount/content/boxcount/html/demo.html
37. **Chang C.-C., and Lin C.-J.,** LIBSVM: a library for support vector machines, 2001. URL: http://www.csie.ntu.edu.tw/_cjlin/libsvm.
38. **Fawcett T.** An introduction to ROC analysis // Pattern Recognition Letters. 2006. V. 27. N 8. P. 861—874.
39. Кривая ошибок. URL: http://www.machinelearning.ru/wiki/

УДК 621.391.16

А. С. Вялых, аспирант,
e-mail: alexandervyalih@gmail.com,
С. А. Вялых, канд. техн. наук, доц.,
А. А. Сирота, д-р техн. наук, проф.,
Воронежский государственный университет

Оценка уязвимости информационной системы на основе ситуационной модели динамики конфликта

Предлагаются возможные подходы к моделированию состояний безопасности информационной системы, которые позволяют учесть динамику изменения ее уязвимостей, квалификацию злоумышленника, а также ряд других параметров, определяющих ситуационный характер конфликтного взаимодействия сторон.

Ключевые слова: *информационная система, злоумышленник, целенаправленная атака, уязвимость, марковская цепь, имитационная модель.*

Введение

В современных условиях функционирование любых технических и информационных систем характеризуется наличием различного рода конфликтных взаимодействий. Участие систем в конфликте определяет не только их облик, но и оказывает существенное влияние на принимаемую политику создания новых систем, развитие информационных технологий и элементной базы. Один из главных вопросов, всегда возникающих в связи с исследованием взаимодействий систем в конфликтной постановке, заключается в оценке потенциальных возможностей достижения успеха сторонами — участниками конфликта. Для оценки надежности и защищенности информационной системы (ИС) требуется не только рассмотрение всех существенных параметров и характеристик в динамике функционирования, но и оценка возможностей противоборствующей стороны — злоумышленников, атакующих ИС. При этом в качестве универсальной синтетической методологии исследований в сфере безопасности ИС целесообразно рассматривать методологию

математического и компьютерного моделирования динамики конфликта, опирающуюся на концептуальные модели конфликтных взаимодействий.

Среди наиболее известных моделей, которые в настоящее время используются для оценки защищенности ИС, можно выделить следующие:

- базовую модель безопасности персональных данных при их обработке в ИС [1], основанную на статическом описании угроз, которым может подвергнуться ИС;
- модель А. Ю. Щеглова [2], описывающую динамику появления уязвимостей в ИС, основанную на теории массового обслуживания.

Однако данные модели обладают рядом ограничений. Так, модель [1] в принципе не учитывает динамику изменений состояний ИС и, в частности, не отражает процессы обнаружения и устранения уязвимостей в ИС, а модель [2] рассматривает данные процессы без привязки к действиям злоумышленников. На практике реализация угроз и действия злоумышленников носят сложный, зависящий от складывающейся ситуации конфликтный характер, что также не учитывается моделями [1—2]. Порядок действий как защищающейся, так и воздействующей сторон в первую очередь определяют такие важные факторы как:

- квалификация и осведомленность злоумышленника об атакуемой системе;
- квалификация и действия (бездействие) администраторов системы и служб технической поддержки;
- характер и длительность переходных процессов при закрытии уязвимостей;
- принимаемые организационные меры и используемые средства защиты информации.

Предлагаемый ниже подход позволяет не только учесть названные факторы, но и дает возможность более полно описывать динамику развития конфликта.

Общая постановка задачи

Пусть имеется ИС с установленным программным обеспечением (ПО). Злоумышленник, целенаправленно атакующий конкретную систему, как правило, проводит предварительную компьютерную разведку ПО, установленного на ИС, исследует уязвимости в этом программном обеспечении и возможные способы использования этих уязвимостей [3, 4]. При этом злоумышленник может обладать различной квалификацией и возможностями.

Рассмотрим математическую модель конфликта, основанную на представлении процесса смены состояний объединенной системы ИС: злоумышленник в виде марковской цепи с конечным числом состояний, переходы между которыми осуществляются по экспоненциальному (пуассоновскому) закону распределения. Данная модель является расширением известной модели [2] в плане учета действий злоумышленника в зависимости от его осведомленности и квалификации. На рис. 1 представлены состояния, в которых может находиться злоумышленник при подготовке и проведении атаки на ИС, а также возможные переходы из одного состояния в другое.

Узлы цепи соответствуют следующим состояниям: S_0 — у злоумышленника отсутствует какая-либо информация о системе; S_1 — у злоумышленника есть информация о ПО ИС; S_2 — у злоумышленника есть информация о ПО ИС и хотя бы об одной уязвимости в этом ПО; S_3 — у злоумышленника есть информация о ПО ИС, хотя бы об одной уязвимости в этом ПО, а также о способе использования хотя бы одной известной ему уязвимости для осуществления атаки на ИС.

Вероятности нахождения в указанных состояниях обозначим соответственно P_0, P_1, P_2, P_3. При этом часть выделенных состояний (S_0, S_1, S_2) агрегируются в состояние "ИС защищена", а состояние S_3 соответствует ситуации "ИС под угрозой".

Предполагается, что злоумышленник может последовательно добывать информацию о ПО ИС, об уязвимостях в этом ПО и далее — о способах использования уязвимостей для осуществления атаки. Интенсивности этих процессов соответственно равны λ_1, λ_2, λ_3. После того как злоумышленник получает всю необходимую информацию, он успешно атакует систему. Администратор ИС, в свою очередь, может изменять используемое в ИС ПО и устранять уязвимости в этом ПО. Интенсивности этих процессов соответственно равны μ_1 и μ_2. Интенсивность устранения известной хакеру уязвимости равна $\mu_2/N_{\text{уязв}}$, где $N_{\text{уязв}}$ — среднее число уязвимостей в системе. Далее предполагается, что при изменении ПО ИС вся информация, которой владел злоумышленник, становится неактуальной, т. е. он возвращается в первоначальное состояние (рассматривается крайний случай, на самом деле информация становится неактуальной частично). При устранении уязвимостей из ПО становится неактуальной информация об этих уязвимостях и способах их использования. Считается, что для марковской цепи установился стационарный режим, поэтому вероятности каждого состояния не

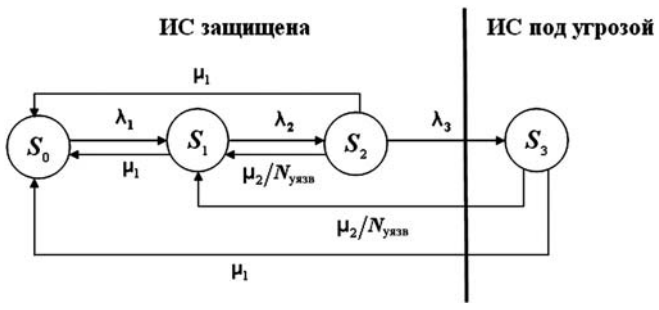


Рис. 1. Модель взаимодействия злоумышленника и атакуемой ИС

зависят от времени. Вероятности каждого из них можно определить, решив систему уравнений [5]

$$\begin{cases} -P_0\lambda_1 + P_1\mu_1 + P_2\mu_1 + P_3\mu_1 = 0; \\ P_0\lambda_1 - P_1(\lambda_2 + \mu_1) + P_2\mu_2/N_{\text{уязв}} + P_3\mu_2/N_{\text{уязв}} = 0; \\ P_1\lambda_2 - P_2(\lambda_3 + \mu_1 + \mu_2/N_{\text{уязв}}) = 0; \\ P_2\lambda_3 - P_3(\mu_1 + \mu_2/N_{\text{уязв}}) = 0; \\ P_0 + P_1 + P_2 + P_3 = 1. \end{cases} \quad (1)$$

ИС защищена, когда общая система "ИС — злоумышленник" находится в состояниях S_0, S_1, S_2, следовательно, вероятность защищенности ИС можно рассчитать по формуле

$$P_{\text{защ}} = P_0 + P_1 + P_2 = 1 - P_3, \quad (2)$$

где P_3 определяется из системы уравнений (1).

Интенсивностям переходов в цепи Маркова (рис. 1) соответствуют следующие временные характеристики (далее, где это особо не оговорено, значения времени приводятся в днях):

- среднее время получения информации об установленном ПО $T_{\text{по}} = 1/\lambda_1$;
- среднее время получения информации об одной уязвимости в ПО $T_{\text{уязв}} = 1/\lambda_2$;
- среднее время получения информации о способе использования уязвимости $T_{\text{сп}} = 1/\lambda_3$;
- среднее время замены ПО $T_{\text{зам\_по}} = 1/\mu_1$ (для определенности данный параметр можно положить равным 730 дням);
- среднее время устранения одной уязвимости $T_{\text{закр\_уязв}} = 1/\mu_2$.

По сравнению с моделью [2] в рассматриваемой модели введен процесс, отражающий действия злоумышленника, а именно, получение информации об ИС, необходимой для ее взлома. Представим по аналогии с [2] процессы обнаружения и устранения уязвимостей в виде системы массового обслуживания с бесконечным числом каналов, тогда среднее число уязвимостей в системе $N_{\text{уязв}}$ можно рассчитать по формуле

$$N_{\text{уязв}} = N_{\text{обн}} T_{\text{закр\_уязв}}/365,$$

где $N_{\text{обн}}$ — число уязвимостей, обнаруженных в ИС за год.

Далее рассматриваются два класса систем: типовая и защищенная. Первая (типовая) система полностью состоит из продуктов Microsoft (например, рабочая станция с установленными операционными системами Windows XP или Windows 7 и типовым набором программ, входящих в установочный пакет, а также с пакетом программ Microsoft Office 2010). Исходя из статистики [6] число обнаруженных в ней за год уязвимостей полагается равным $N_{\text{обн}} = 500$. Для второй (защищенной) системы [7] предполагается, что $N_{\text{обн}} = 10$. Системы могут быть атакованы злоумышленниками разных уровней квалификации. Для примера рассмотрим злоумышленников, обладающих следующими способностями: профессионал ($T_{\text{по}} = 5$, $T_{\text{уязв}} = 1$, $T_{\text{сп}} = 1$); средний злоумышленник ($T_{\text{по}} = 10$, $T_{\text{уязв}} = 5$, $T_{\text{сп}} = 5$); слабый злоумышленник ($T_{\text{по}} = 20$, $T_{\text{уязв}} = 10$, $T_{\text{сп}} = 10$). Следует отметить, что рассматриваемый здесь слабый злоумышленник, чтобы обеспечить указанные параметры своего воздействия, должен при этом являться специалистом высокого уровня в области информационных технологий.

Заданные параметры системы и злоумышленника позволяют решить систему уравнений (1), (2) и в явном виде определить аналитические зависимости вероятности защищенности каждой из рассматриваемых ИС от $T_{\text{закр\_уязв}}$ — среднего времени устранения одной уязвимости.

Компьютерное моделирование динамики ситуационного конфликта

Рассмотренная математическая модель в виде марковской цепи дает возможность оценить только среднестатистические характеристики пребывания в каждом состоянии для установившегося режима и не позволяет оценить динамику переходных процессов в зависимости от складывающейся ситуации и используемых сторонами стратегий конфликтного противоборства. Очевидно, например, что одной из доминирующих стратегий сторон в рассматриваемой ситуации является стремление выполнить упреждающие по времени действия по отношению к противоборствующей стороне. При этом распределение времени переходов в различные состояния носит произвольный, отличающийся от пуассоновской модели, характер. Кроме того, часто возникает необходимость рассматривать ситуацию, принципиально отличающуюся от дуэльной, когда конфликт затрагивает несколько участников с каждой стороны (например, ИС атакуют не один, а несколько злоумышленников).

Усложнение постановки задачи и необходимость учета всех значимых для описания информационного конфликта факторов неминуемо ведут к возрастающим трудностям при использовании аналитических математических моделей. Это определяет существенную роль средств и компьютерных технологий объектно-ориентированного моделирования для исследования закономерностей конфликта. Одним из доступных компьютерных средств и естественным для описания динамики ситуационного конфликта механизмом реализации компьютерных имитационных моделей информационного конфликта систем является, на наш взгляд, использование формализма гибридных автоматов (карт состояний Харела) и тех возможностей, которые для этих целей предоставляет интегрированная среда MATLAB + Simulink + Stateflow [8].

Взаимодействие злоумышленника и ИС в терминах работы [8] можно описать с помощью SF-модели, которая имеет вид, показанный на рис. 2. Модель состоит из трех параллельно функционирующих объектов (*Sysadmin* и *System* с одной стороны, *Haker* с другой стороны), в которых разме-

щены карты состояний, описывающие возможные значения учитываемых факторов и поведение (в зависимости от этих значений) всех участвующих в конфликте.

В отличие от моделей конфликта, рассмотренных в работе [8], в представленной модели ни одна из сторон не может добиться абсолютной победы, т. е. в случае перехода ИС в незащищенное состояние она может снова вернуться в защищенное состояние (восстановиться). Поэтому в ходе эксперимента будет рассчитываться не число побед сторон конфликта (например, вероятность перехода ИС в незащищенное состояние), а вероятность нахождения ИС в защищенном состоянии.

Информационная система (блок System) может находиться в двух основных состояниях:

- состояние *Zashishena* — состояние, при котором ИС считается защищенной от атак злоумышленника;
- состояние *Nezashishena* — состояние, при котором ИС считается незащищенной от атак злоумышленника.

Блок *Sysadmin* состоит из двух параллельно функционирующих элементов *Po* и *Uyazv*, имитирующих соответственно действия администратора ИС по смене ПО и устранению уязвимостей. Параллельность их работы объяснятся тем, что чаще всего устранение уязвимостей из ПО происходит за счет установки обновлений, предоставляемых разработчиками данного ПО, и при установке нового ПО системный администратор может сразу установить обновление на это ПО.

Системный администратор в процессе замены ПО (модуль *Po*) может находиться в двух состояниях:

- состояние *Podgotovka* — состояние, при котором администратор ИС готовится к замене ПО;
- состояние *SmenaPo* — состояние, при котором администратор ИС меняет ПО.

Упреждающий переход в последнее состояние приводит к генерации события *zam_po*, которое переводит блок *Haker* в состояние *Net_informacii* (вся информация, которой на тот момент обладал злоумышленник, теряет актуальность). Одновременно блок *System* переходит в состояние *Zashishena*, что соответствует переходу ИС в защищенное состояние.

Из состояния *SmenaPo* подблок *Po* сразу же переходит в состояние *Podgotovka* (считается, что замена ПО происходит мгновенно).

При устранении уязвимостей (модуль *Uyazv*) системный администратор может находиться в двух состояниях:

- состояние *Podgotovka* — состояние, при котором администратор ИС готовится к закрытию уязвимости в ПО;
- состояние *ZakrUyazv* — состояние, при котором администратор ИС закрывает уязвимость в ПО, известную хакеру.

Упреждающий переход в последнее состояние приводит к генерации события *zakr_u*, которое переводит блок *Haker* из любого состояния, кроме *Net_informacii,* в состояние *Informaciya_o_PO* (информация об уязвимостях и способах взлома, которой на тот момент обладал злоумышленник, теряет актуальность). При этом блок *System* переходит в состояние *Zashishena*, что соответствует переходу ИС в защищенное состояние.

Из состояния *ZakrUyazv* подблок *Uyazv* сразу же переходит в состояние *Podgotovka* (считается, что закрытие уязвимости в ПО происходит мгновенно).

Состояния, в которых может находиться сторона *Haker*, полностью соответствуют состояниям злоумышленника, определенным в рассмотренной выше марковской модели взаимодействия злоумышленника и ИС:

- состояние *Net_informacii* — начальное состояние, при котором у злоумышленника отсутствует какая-либо информация о системе;
- состояние *Informaciya_o_PO* — состояние, при котором у злоумышленника есть информация о ПО ИС;
- состояние *Informaciya_o_uyazvimostyah* — состояние, при котором у злоумышленника есть информация о ПО ИС и хотя бы об одной уязвимости в этом ПО;
- состояние *Invormaciya_o_sposobah_vzloma* — состояние, при котором у злоумышленника есть информация о ПО ИС, хотя бы об одной уязвимости в этом ПО, а также информация о способе использования хотя бы одной известной ему уязвимости для осуществления атаки на ИС.

При упреждающем достижении последнего состояния происходит генерация события *vzlom*, пе-

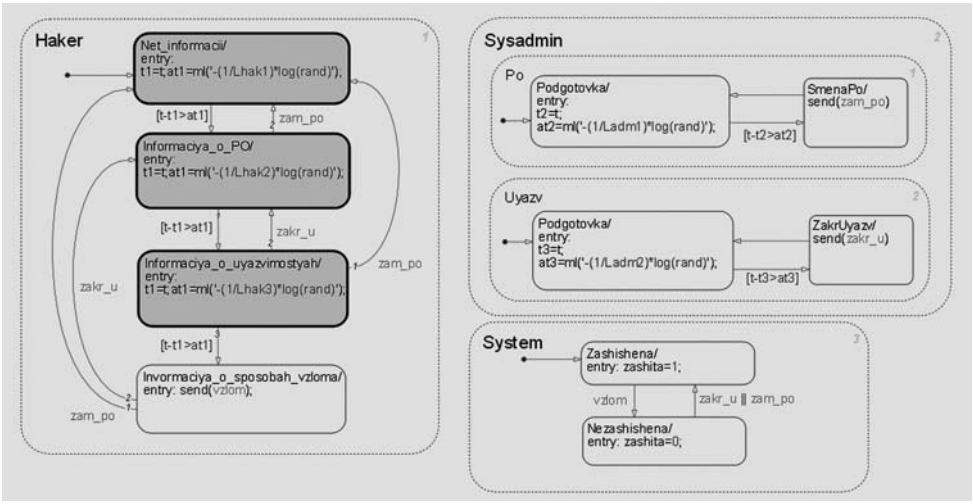


Рис. 2. SF-модель конфликта

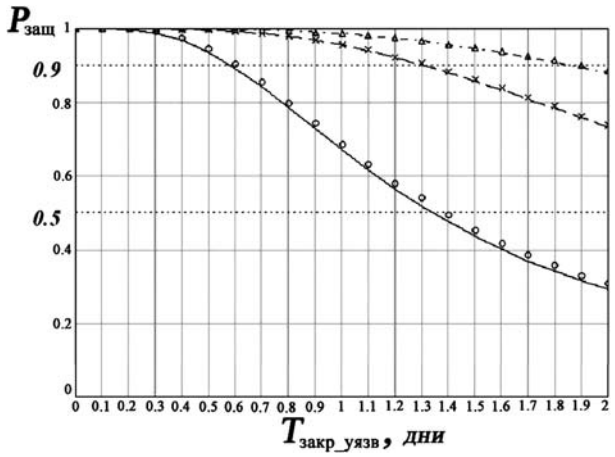


Рис. 3. Вероятность защищенности типовой системы от атак профессионалов, средних и слабых злоумышленников (5 лет)

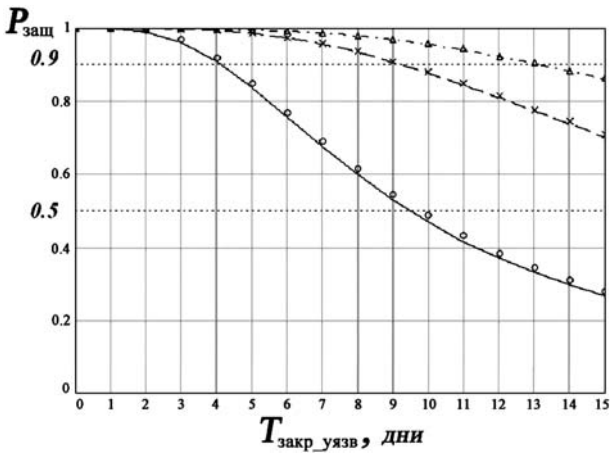


Рис. 4. Вероятность защищенности защищенной системы от атак профессионалов, средних и слабых злоумышленников (5 лет)

реводящего блок *System* в состояние *Nezashishena*, что соответствует переходу ИС в незащищенное состояние.

Время перехода сторон *Haker, Sysadmin* и *System* в любое из возможных состояний описывается переменными $t1$, $t2$ и $t3$ соответственно. Время пребывания злоумышленника (стороны *Haker*) в каждом из состояний в отсутствие событий *zam_po* и *zakr_u at*1 является случайным и задается путем вызова *m*-функции $-(1/Lambda)\cdot\log(rand)$, формирующей случайное число, распределенное по показательному закону с параметром злоумышленника *Lambda* ($Lhak1 = 1/T_{по}$, $Lhak2 = 1/T_{уязв}$ или $Lhak3 = 1/T_{сп}$ в зависимости от состояния), при этом переходы из одного состояния в другое осуществляются по условию истечения времени пребывания в каждом из состояний. Время подготовки к замене ПО *at*2 и подготовки к устранению известных хакеру уязвимостей *at*3 определяется аналогичным способом. Следует отметить, что принципиальных ограничений на вид законов распределения в данной модели не существует.

В имитационной модели используются параметры злоумышленника и ИС такие же, как и в рассмотренной выше марковской модели. Время эксперимента берется равным пяти годам (1825 дням), т. е. полученная оценка вероятности защищенности ИС будет актуальна для долгосрочной перспективы. Шаг дискретизации берется равным 0,1 дня. Необходимое число испытаний, в соответствии с работой [8], выбирается равным $N\_isp = 1000$.

Для подсчета вероятности защищенности системы при конкретном времени закрытия одной уязвимости требуется моделирование тысячи вариантов реализации исследуемого процесса, что на компьютере с процессором AMD Athlon(tm) 64 Processor 3200+ занимает 15 мин. На рис. 3 и 4 приведены зависимости вероятности защищенности ИС от времени закрытия одной уязвимости $T_{закр\_уязв}$, которые были получены для различных предполагаемых злоумышленников на основе имитационного моделирования и модели динамики конфликта на основе марковской цепи, описанной выше.

Сплошными линиями нанесены вероятности защищенности 1-й и 2-й систем от атак профессионалов, штриховой линией — от атак средних злоумышленников, штрихпунктиром — от атак слабых злоумышленников, полученные при использовании модели динамики конфликта на основе марковской цепи. Кружками, крестиками и треугольниками нанесены те же вероятности, но полученные в ходе эксперимента с помощью имитационной модели.

Разница результатов моделирования составляет менее 0,02. Такое расхождение можно считать несущественным и объяснить переходным периодом, который учитывает имитационная модель и не учитывает модель на основе цепи Маркова. Следовательно, можно говорить об адекватности разработанных моделей при рассмотрении работы ИС в долгосрочной перспективе. На рис. 5, 6 приведены результаты, аналогичные рассмотренным выше, для более короткого срока моделирования, описывающие работу новой ИС в течение двух месяцев (60 дней).

Разница между результатами моделирования уже составляет примерно 0.1, что для оценки вероятности защищенности ИС представляется существенным. Следовательно, при оценке защищенности системы в переходные периоды, например, при вводе ее в эксплуатацию, предпочтительней использовать средства компьютерного имитационного моделирования.

Изложенные приемы визуального программирования модели конфликта систем в дуэльной ситуации могут быть положены в основу создания более сложной модели конфликта группировок (коалиций) систем (например, в случае распределенных атак). Основные трудности, возникающие при построении подобных моделей, состоят в необходимости воспроизведения механизма управления коалициями систем и связанного с ним механизма изменения взаимосвязей и состава противоборствующих сторон

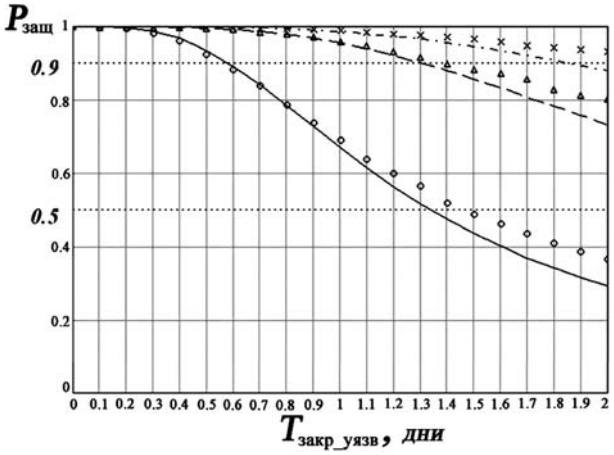


Рис. 5. Вероятность защищенности типовой системы от атак профессионалов, средних и слабых злоумышленников (два месяца)

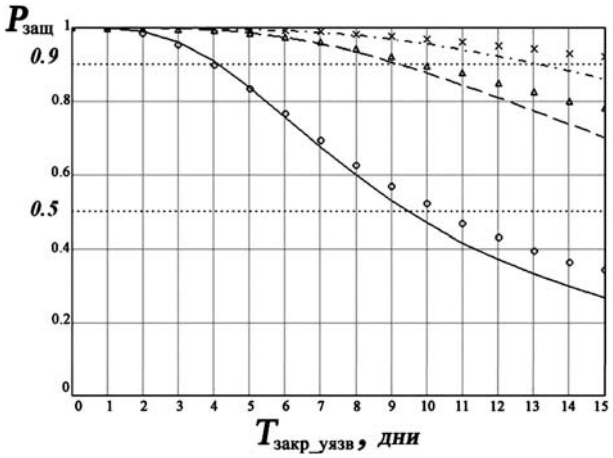


Рис. 6. Вероятность защищенности защищенной системы от атак профессионалов, средних и слабых злоумышленников (два месяца)

в динамике конфликта. В частности, потребуется моделировать эффекты "перенацеливания" отдельных участников конфликта после того, как они достигают частных выигрышей в дуэльных ситуациях.

Одним из преимуществ разработанной ситуационной модели динамики конфликта является возможность учета угроз от так называемых внутренних злоумышленников. Для рассмотрения таких угроз достаточно лишь выбрать соответствующие параметры злоумышленника. Например, можно считать, что злоумышленник уже владеет информацией о ПО ИС, об уязвимостях в этом ПО, а для поиска информации о способах использования этих уязвимостей ему требуется мало времени (например, менее одного дня), что в представленной модели определяется заданием следующих значений: $T_{по} = 0$, $T_{уязв} = 0$, $T_{сп} = 0{,}5$.

Выводы

Представленные модели ИС различных классов не претендуют на точное соответствие реальности абсолютных значений оцениваемых параметров, так как необходимые исходные данные (характеристики ИС, администраторов ИС и злоумышленников, которые могут атаковать ИС), должны в каждом конкретном случае оцениваться соответствующими экспертами. Тем не менее, полученные результаты моделирования позволяют сформулировать определенные рекомендации для системных администраторов, обслуживающих рассмотренные ИС.

Для достижения вероятности защищенности типовой и защищенной системы от слабого злоумышленника не менее 0,9 необходимо устранять из ИС уязвимость каждые 13 и каждые 1,8 дней соответственно. В то же время данные, опубликованные в работах [6, 7], свидетельствуют о том, что время устранения одной уязвимости из ИС в периоды между обновлениями ПО в разы превышает полученные значения (обновления, закрывающие уязвимости, выходят приблизительно раз в месяц, что соответствует вероятности защищенности от слабого злоумышленника для типовой системы менее 0,1, а для защищенной приблизительно 0,5). Следовательно, можно говорить о том, что современные системы практически не защищены от рассмотренных атак.

Чтобы повысить защищенность ИС, рекомендуется не только своевременно устанавливать обновления ПО, но и использовать дополнительные средства (предпринимать специальные меры) поиска и устранения уязвимостей из ИС. Кроме этого, необходимо обязательно использовать дополнительные средства защиты ИС, например, межсетевые экраны (программные или аппаратные), средства доверенной загрузки и разграничения доступа, которые позволяют значительно повысить защищенность систем, использующих заведомо проблемные службы и информационные технологии с большим числом уязвимостей.

Исследование влияния указанных факторов также целесообразно проводить на основе средств визуального объектно-ориентированного моделирования конфликтного взаимодействия систем, использующих формализм гибридных автоматов.

Список литературы

1. **Базовая** модель угроз безопасности персональных данных при их обработке в информационных системах персональных данных (выписка). URL: http://www.fstec.ru/_spravs/model.rar

2. **Щеглов А. Ю.** Безопасность современных ОС "в цифрах" URL: http://www.itsec.ru/articles2/Inf_security/bezopasnost-OS

3. **Лукацкий А. В.** Обнаружение атак. С-Пб, БХВ-Петербург, 2001. 624 с.

4. **Скембрей Дж., Мак-Клар Ст., Курц Дж.** Секреты хакеров. Безопасность сетей — готовые решения. М.: Вильямс, 2001. 656 с.

5. **Кемени Джон Дж., Дж. Лори Снелл.** Конечные цепи Маркова. М.: Наука, 1970. 272 с.

6. **Microsoft** Security Intelligence Report v9. URL: http://www.microsoft.com/security/sir/

7. **Vulnerability** Report: Cisco IOS 12.x. URL: http://secunia.com/advisories/product/182/?task = statistics_2009

8. **Алгазинов Э. К., Сирота А. А.** Анализ и компьютерное моделирование информационных процессов и систем / Под общ. ред. А. А. Сироты. М.: Диалог-МИФИ, 2009. 416 с.

УДК 004.32

В. А. Колосков, д-р техн. наук, проф.,
Г. П. Колоскова, канд. техн. наук, проф.,
Динь Туан Лонг, аспирант,
"МАТИ" — РГТУ имени К. Э. Циолковского,
e-mail: v_koloskov@mail.ru

Управляемая клеточная непрерывная среда самореконфигурации многопроцессорных систем

Представлен подход к самореконфигурации в однородных структурах многопроцессорных систем (МПС) на основе клеточной среды реконфигурации. Дан клеточный алгоритм реконфигурации отказоустойчивой МПС, использующий модель естественно-подобной среды для параллельного поиска маршрутов восстановления. Рассмотрены правила обработки локальных данных, обеспечивающие восстановление логической структуры сети при многократных отказах.

Ключевые слова: *многопроцессорные системы, отказоустойчивость, реконфигурация, клеточный алгоритм*

Введение

Важной задачей разработки многопроцессорных систем (МПС) критического применения является обеспечение их отказоустойчивости. Для восстановления работоспособности в отказоустойчивых МПС при появлении отказов выполняется ее реконфигурация [1—3], состоящая в перераспределении программных модулей (ПМ) между работоспособными элементами системы, включая резервные. Определение исполняемых ПМ в процессорных элементах (ПЭ) выполняется на основании глобальных данных о размещении отказавших и резервных элементов. Для восстановления системы необходима перезагрузка исполняемых ПМ по новым адресам либо перенастройка ПЭ на новые программные модули при хранении всех копий в памяти каждого процессора. При этом для перезагрузки исполняемых ПМ по новым адресам требуется прерывание работы системы, а изменение размерности МПС влечет за собой перепроектирование алгоритма реконфигурации.

Авторами разработан и исследуется универсальный подход [4—8] к реконфигурации типовых решетчатых структур с размещенными в них резервными элементами. Подход основан на автоматической автономной клеточной перенастройке (самореконфигурации) минимального подмножества элементов МПС на ПМ соседних элементов. Самореконфигурация реализуется клеточной средой с решетчатой топологией, каждая ячейка которой связана с одним, только ей принадлежащим ПЭ, а каждый ПЭ хранит постоянный состав копий ПМ смежных (по физическим связям) элементов, что позволило исключить перезагрузку программ. Реакцией клеточной среды на отказы является образование в ней устойчивых структур, содержащих информацию о реконфигурации МПС. Ячейки клеточной среды при формировании структур параллельно выполняют неизменный набор клеточных операций обработки минимальных локальных данных о состоянии соседних узлов. Локальные данные не содержат глобальных сведений о размерах решетки, о размещении в решетке отказавших и резервных элементов. Кооперативное взаимодействие ячеек среды обеспечивает сохранение непрерывности работы при появлении отказов и реконфигурационной способности системы при ее масштабировании.

В настоящей статье представлен универсальный клеточный алгоритм реконфигурации, реализуемый на базе модели управляемой токопроводящей среды. Приведены клеточные правила обработки локальных данных о состоянии узлов сети. Показан механизм получения решения по самореконфигурации структуры МПС путем управления связями в естественно-подобной клеточной среде.

Объект исследования

Структура отказоустойчивой многопроцессорной системы представляется ортогональной графовой решеткой из $m \times n$ узлов (вершин), где m — число узлов по оси ординат, n — число узлов по оси абсцисс. При свертывании границ решетки по вертикали и горизонтали она превращается в тор. Расстояние между соседними вершинами по всем $k \in (1, 2, 3, 4)$ (1 — вправо, 2 — вверх, 3 — вниз, 4 —влево) направлениям постоянно и равно шагу решетки d, расстояние между удаленными друг от друга вершинами определяется в ортогональной метрике. Процессорный элемент МПС является универсальным элементом замены, позволяющим перестраивать его на функции любого из четырех соседних элементов. При этом рабочий ПЭ хранит собственный ПМ и копии программных модулей

соседних ПЭ, а резервный элемент — только копии соседей. Хранение копий ПМ соседей позволяет оперативно без задержек на перезагрузку ПМ в реконфигурированной после отказа МПС восстанавливать функции элементов. Резервный ПЭ МПС служит для восстановления исходного числа рабочих работоспособных элементов и заменяет один из соседних ПЭ, используя ПМ этого элемента.

Каждый ПЭ имеет физический адрес, постоянно закрепленный за ним, и логический адрес, характеризующий исполняемый элементом ПМ. В начальном состоянии физический и логический адреса ПЭ совпадают. Логические адреса обеспечивают независимость ПМ от физических ресурсов, на которых они выполняются. В случае отказов элементов с помощью клеточной реконфигурации выполняется изоляция неисправных ПЭ, включение в работу необходимого числа резервных ПЭ и настройка работоспособных элементов на новые логические адреса ПМ. Клеточная реконфигурация реализуется клеточной средой, распределяемой в решеточной структуре МПС (рис. 1, *а*). При этом каждый ПЭ (P_{ij}) с резервными копиями ПМ соседних элементов (рис. 1, *б*) имеет собственную ячейку среды реконфигурации (C_{ij}). По локальной информации от физических соседей и собственному состоянию ячейка реконфигурации сохраняет логический адрес в $P_{i,\,j}$ либо изменяет его на логический адрес одного из соседей $\{P_{i,\,j+1},\ P_{i+1,\,j},\ P_{i,\,j-1},\ P_{i-1,\,j}\}$.

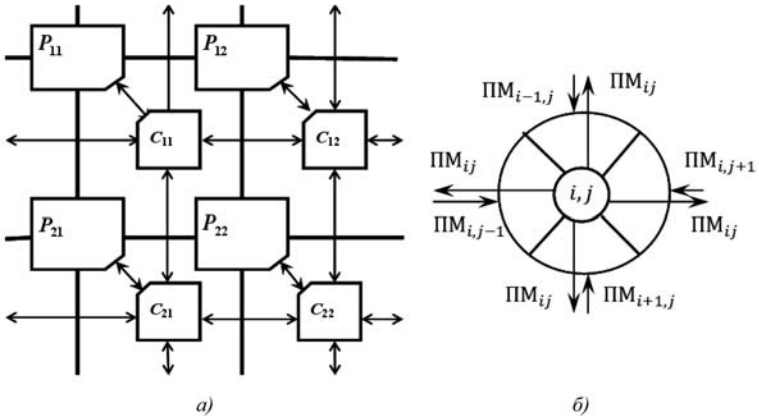


Рис. 1. Структура самореконфигурируемой МПС (*а*) и модель загрузки ПЭ копиями ПМ (*б*)

Интеграция ячейки клеточной среды C_{ij} с ПЭ P_{ij} позволяет строить универсальный самонастраиваемый элемент замены для соседних элементов, который в совокупности с такими же элементами МПС обеспечивает самореконфигурацию при смене комбинаций отказавших элементов независимо от варианта размещения резервных элементов и размерности решетки МПС. Загрузка программных модулей и их копий, а также настройка P_{ij} на основную или резервную функции выполняются перед началом работы сети, отказы в сети обнаруживаются средствами контроля элементов, инициирующих процесс самореконфигурации. Целостность структуры МПС обеспечивается встроенной сетью маршрутизаторов, реализующей адаптированный алгоритм поиска ПМ в решетке МПС с отказавшими элементами.

Формулировка задачи

В результате реконфигурации число работоспособных элементов сохраняется неизменным за счет включения требуемого числа резервных узлов S_{R1} вместо отказавших S_O ($|S_o| = |S_{R1}|$). В процессе реконфигурации формируется новое множество S_{A1} активизированных работоспособных ПЭ: $S_{A1} = (S_A \backslash S_O) \cup S_{R1}$.

Задача реконфигурации заключается в поиске отображения множества S_L логических адресов на множество S_{A1} активизированных работоспособных ПЭ: $S_L \to S_{A1}$, при котором число B перенастраиваемых (изменяющих логические адреса) ПЭ для произвольных конфигураций отказавших и резервных элементов стремится к минимуму:

$$B \to \min. \qquad (1)$$

Минимизация числа перенастраиваемых ПЭ позволяет для большинства элементов сохранять неизменным время поиска приемника при передаче сообщений в реконфигурируемой МПС.

Разработка способа клеточной реконфигурации связана с решением следующих задач:

- определение минимального подмножества работоспособных элементов для участия в реконфигурировании;
- формирование подмножества резервных ПЭ для восстановления исходного числа рабочих элементов;
- восстановление логической структуры на множестве работоспособных элементов в соответствии с критерием (1).

Решение перечисленных задач выполняет клеточная среда реконфигурации МПС. При этом для удовлетворения критерию (1) в направлениях физических связей между элементами вычисляются кратчайшие непересекающиеся маршруты от отказавших элементов к резервным узлам. Множество элементов маршрутов определяет минимальное множество ПЭ, изменяющих логические адреса. Конечные пункты маршрутов соответствуют ближайшим к местам отказов резервным элементам и обеспечивают восстановление исходного числа работоспособных ПЭ. Распределение логических адресов реализуется настройкой каждого ПЭ маршрута на функцию своего предшественника, что заменяет перезагрузку системы автоматической заменой исполняемой резервной копии ПМ элемента.

Реконфигурация на базе управляемой естественно-подобной среды

Для снижения трудоемкости проектирования и функциональной сложности специализированной вычислительной среды реконфигурации целесообразно использовать естественно-подобную среду, физические законы функционирования которой позволяют получать значения требуемых величин.

В настоящей работе для параллельного вычисления непересекающихся маршрутов применена управляемая токопроводящая решетка, обеспечивающая получение непрерывных значений характеристик длин маршрутов. Узел решетки (рис. 2) в каждом направлении $k = \overline{1,4}$ включает резистор R для снятия значения тока и ключ связи с соседним узлом sw_k^{ij}. Состоянием ключа sw_k^{ij} (замкнуто/разомкнуто) управляет сигнал y_k^{ij}. Общая точка узла, связывающая резисторы R, через ключи sw_o^{ij}, sw_r^{ij}, соединена с потенциалами E и 0 соответственно. Если процессорный элемент P_{ij} отказал, то узел получает потенциал E, если P_{ij} используется в качестве резервного, то узел подключается к нулевому потенциалу. Значения I_k^{ij} и φ_k^{ij} характеризуют ток и потенциал на k-м выходе узла.

Закономерности функционирования представленной среды позволяют получать значения характеристик удаленности от резервных элементов для любого узла решетки. Построение маршрутов в решетке базируется на следующем утверждении.

Утверждение. *Направление максимального тока, вытекающего из узла, соответствует направлению кратчайшего маршрута к резервному элементу.*

Из утверждения следует, что максимальный ток определяет направление минимальной удаленности от узла с нулевым потенциалом. Поскольку нулевой потенциал имеют узловые точки резервных элементов, то направление максимального вытекающего тока соответствует направлению маршрута к ближайшему резервному элементу.

На рис. 3 представлен пример распределения потенциалов и токов в решетке с отказавшим элементом в позиции $(i, j) = (3,2)$ и двумя резервными элементами в позициях (2,1) и (1,4). По максимальным значениям вытекающих токов (15,8 и 9,2) из узлов (3,2) и (3,1) достигается ближайший резервный элемент по кратчайшему маршруту. В ортогональной метрике длина такого маршрута равна двум шагам решетки.

Если для нескольких отказавших узлов ближайшим оказывается один и тот же резервный элемент, возможны пересечения маршрутов. Так, для отказавших элементов (3,2) и (3,3) максимальные значения вытекающих токов приводят к одному резервному элементу (2,2) (рис. 4). Появление подобных ситуаций требует корректировки связей токопроводящей решетки для устранения конфликтов минимальных маршрутов.

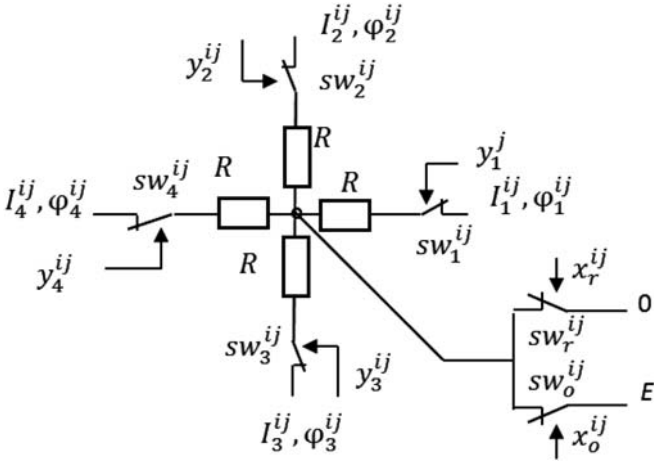


Рис. 2. Структура узла токопроводящей решетки

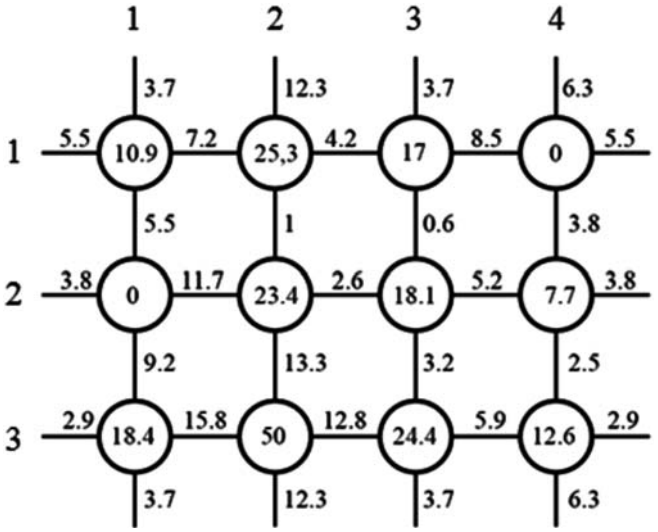


Рис. 3. Пример распределения токов и потенциалов

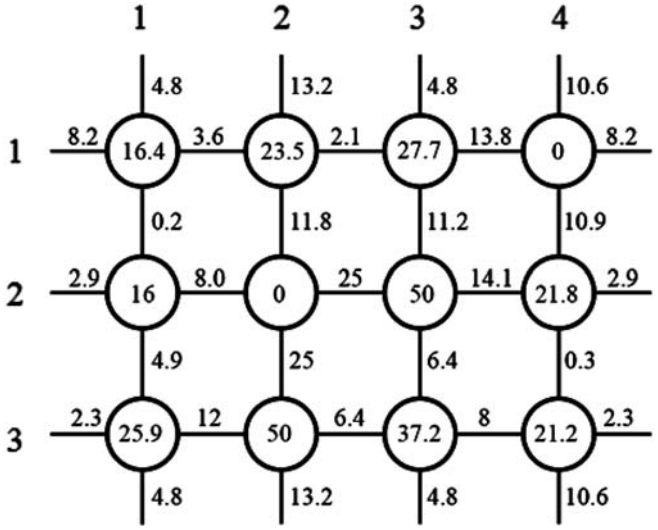


Рис. 4. Пример конфликтного распределения потенциалов

Управление связями в модели токопроводящей среды позволяет адаптироваться к появлению столкновений маршрутов и обеспечить построение минимальных непересекающихся маршрутов от мест отказов к резервным элементам. Переназначение логических адресов выполняется для минимального множества работоспособных элементов, принадлежащих маршрутам.

В работе исследован асинхронный подход к построению минимальных маршрутов реконфигурации, в соответствии с которым предшественники и последователи маршрута выбираются в процессе итерационных вычислений. При этом корректируются только конфликтные связи столкновения маршрутов по мере их появления.

Асинхронный клеточный алгоритм реконфигурации

Клеточный асинхронный алгоритм реконфигурации представляется системой параллельных замещающих операций $\Theta = \{\Theta_1, \Theta_2, \Theta_3, \Theta_4, \Theta_5, \Theta_6\}$, где операции Θ_1, Θ_2 моделируют процесс получения устойчивых значений токов и потенциалов в управляемой токопроводящей решетке, операции Θ_3, Θ_4, Θ_5 обеспечивают управление состоянием ключей токопроводящей решетки и перенастройкой логических адресов ПЭ, Θ_6 фиксирует фатальный отказ МПС.

Система Θ описывает итерационные операции обработки клеточного массива $\{S_{ij}\}$ ($i = 1, 2, ..., m$; $j = 1, 2, ..., n$). Слово состояния S_{ij} любой клетки для разработанного алгоритма определяется значениями множества переменных $S_{ij} = \{x_{\text{o}}^{ij}, x_r^{ij}, \Phi^{ij}, I^{ij}, M^{ij}, Y^{ij}\}$, где $x_{\text{o}}^{ij} \in \{1, 0\}$ — отказ/работоспособность узла (i,j); $x_r^{ij} \in \{1, 0\}$ — резервный/основной элемент; $\Phi^{ij} = \{\varphi_k^{ij}\}$ — множество значений потенциалов узла (i, j) на выходах $k \in (1, 2, 3, 4)$; $I^{ij} = \{I_k^{ij}\}$ — значения токов узла (i, j) по направлениям $k = \overline{1,4}$; $M^{ij} = \{M_k^{ij}\}$ — наличие/отсутствие маршрутов из узла (i, j) в направлениях $k \in (1, 2, 3, 4)$; $Y^{ij} = \{Y_k^{ij}\}$ — переменные состояния ключей (замкнут — 1, разомкнут — 0) по каждому из направлений $k = \overline{1,4}$.

На каждой итерации для всех клеток выполняются все предписанные алгоритмом правила обработки и вырабатывается промежуточное значение клеточного массива. Итоговое состояние клеточного массива после выполнения всех операций Θ представляет результат клеточной реконфигурации МПС для исходной комбинации отказов и заданного размещения резервных элементов.

Правила формирования значений переменных клетки определим с учетом обозначений входных направлений для узла (i, j) переменными $(p, q) \in \{(i, j-1), (i+1, j), (i-1, j), (i, j+1)\}$.

Вычисление установившихся значений потенциалов узла (i, j) по направлениям связей ($k = \overline{1,4}$) для текущего варианта расположения отказавших и резервных узлов выполняется в соответствии с клеточной операцией Θ_1:

$$\Theta_1 : \varphi_k^{ij} = \begin{cases} \varphi_H, \text{ если } y_k^{ij} = 0; \\ E, \text{ если } x_{\text{o}}^{ij} \wedge \bar{x}_r^{ij} \wedge y_k^{ij} = 1; \\ 0, \text{ если } \bar{x}_{\text{o}}^{ij} \wedge x_r^{ij} \wedge y_k^{ij} = 1; \\ \sum\limits_{s=1}^{4} (\varphi_{5-s}^{pq} y_s^{ij} y_{5-s}^{pq}) / \sum\limits_{s=1}^{4} (y_s^{ij} y_{5-s}^{pq}), \\ \text{если иначе,} \end{cases} \quad (2)$$

где φ_H — значение потенциала разрыва связи между вершинами (i, j) и (p, q).

Процесс вычисления завершается, когда ни одна клетка не может изменить свои данные S_{ij}. Процесс образования потенциалов заканчивается после получения значений, удовлетворяющих условию: $\forall (i, j)$: $\varphi^{ij}(t+1) - \varphi^{ij}(t) \leqslant \Delta$, где Δ — заданная погрешность вычислений. Установившиеся значения потенциалов $\{\varphi_k^{ij}\}$ позволяют получить значения токов $\{I_k^{ij}\}$ (при сопротивлениях $R = 1$) узла (i, j) по всем направлениям $k = \overline{1,4}$ в соответствии с операцией Θ_2:

$$\Theta_2 : I_k^{ij} = \begin{cases} \varphi_k^{ij} - \varphi_{5-k}^{pq}, \text{ если } y_k^{ij} \wedge y_{5-k}^{pq} = 1; \\ 0, \text{ если иначе.} \end{cases} \quad (3)$$

Значения переменных токов и потенциалов описывают состояние токопроводящей среды, на базе которой вычисляются минимальные маршруты от мест отказов до ближайших резервных узлов с обходом отказавших элементов. Операция определения направления исходящей из узла (i, j) дуги маршрута ($M_k^{ij} = 1$) к ближайшему резервному элементу имеет вид

$$\Theta_3 : M_k^{ij} = \begin{cases} 1, \text{ если } (I_k^{ij} = \max\limits_{s=\overline{1,4}} \{I_s^{ij}\}) \wedge (I_k^{ij} > 0) \wedge \\ \wedge \left(x_{\text{o}}^{ij} \vee \left(\bigvee\limits_{s=1}^{4} M_{5-s}^{pq} \right) \right); \\ 0, \text{ если иначе.} \end{cases} \quad (4)$$

В соответствии с Θ_3 исходящая дуга маршрута выбирается в направлении максимального исходящего тока. Узел (i, j) может быть начальным ($x_{\text{o}}^{ij} = 1$) или промежуточным пунктом (если узел имеет хотя бы одну входную дугу: $M_1^{i,j-1} \vee M_2^{i+1,j} \vee M_3^{i-1,j} \vee M_4^{i,j+1}$ маршрута либо не принадлежать маршруту. С помощью операции Θ_3 определяется минимальное множество работоспособных элементов, участвующих в реконфигурации и формируется подмножество активизируемых резервных элементов.

При появлении столкновений маршрутов (наличии в узле более одного входного направления с $M_k^{pq} = 1$) сохранятся только один из них, остальные запрещаются. Разрешение конфликтов столкновений обеспечивается корректировкой состояний (размыканием) ключей токопроводящей среды. В качестве критерия выбора одного из нескольких входных маршрутов используется минимум длины маршрута (отрицательный минимальный ток I_k^{ij}), при равноценных вариантах выбирается направление с наибольшим номером (значением переменной k):

$$\Theta_4: y_k^{ij} = \begin{cases} 0, \text{ если } (M_{5-k}^{pq} = 1) \wedge (I_k^{ij} \neq \min\limits_{s=\overline{1,4}} \{I_s^{ij}\}); \\ 1, \text{ если иначе.} \end{cases} \quad (5)$$

Разомкнутое состояние ключей во избежание многократных повторений столкновений сохраняется до окончания процесса реконфигурации. В результате корректировок связей в токопроводящей среде формируются непересекающиеся минимальные маршруты от мест отказов к ближайшим резервным узлам. При этом клеточные вычисления используют только локальные данные о состоянии соседних узлов, не располагая глобальными данными о числе и расположении отказавших и резервных элементов в решетке.

Если маршруты построены не для всех отказавших узлов, то фиксируется фатальный отказ МПС (операция Θ_5):

$$\Theta_5: FO^{ij} = \begin{cases} 1, \text{ если } (x_0^{ij} = 1) \wedge \left(\bigvee\limits_{k=1}^{4} M_k^{ij} = 0 \right); \\ 0, \text{ если иначе.} \end{cases} \quad (6)$$

Процесс самореконфигурации МПС состоит в формирования новых логических адресов $\{L^{ij}\}$ на основе данных о полученных маршрутах:

$$\Theta_6: L^{ij} = \begin{cases} (p, q), \text{ если } M_{5-k}^{pq} = 1; \\ (i, j), \text{ если иначе.} \end{cases} \quad (7)$$

Обобщенный асинхронный алгоритм функционирования клеточной среды реконфигурации включает следующие этапы:

1. Инициализация клеточных переменных по сигналам от средств контроля: $\forall$ ($i = \overline{1, m}$, $j = \overline{1, n}$) $x_o^{ij} \in \{1, 0\}$ — отказ/работоспособность узла (i, j); $x_r^{ij} \in \{1, 0\}$ — резервный/основной элемент; Φ^{ij}, I^{ij}, M^{ij} — установка в нулевые значения; Y^{ij} — установка в единичные значения (ключи замкнуты).

2. Вычисление установившихся значений потенциалов на выходах узла и токов в связях узлов (i, j) ($i = \overline{1, m}$, $j = \overline{1, n}$) с соседними клетками (формулы (2), (3)) для текущего состояния ПЭ и состояний ключей.

3. Вычисление значений переменных маршрутов, управления ключами и фатального отказа (формулы (4)—(6)).

4. Если значения переменных клеточного массива изменились, возврат к п. 2.

5. Если $FO^{ij} = 1$ (формула (6)) хотя бы для одного узла, то фиксация фатального отказа в МПС. Переход к п. 7.

6. Перенастройка логических адресов элементов маршрутов (формула (7)).

7. Конец.

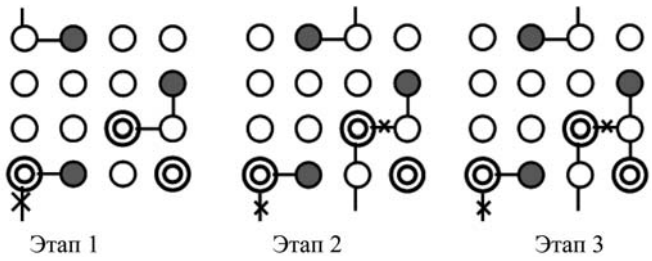


Рис. 5. Эволюция маршрутов реконфигурации

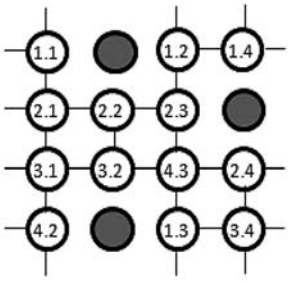
Ниже рассмотрен пример выполнения алгоритма реконфигурации. Для решетки с отказами в позициях (1,2), (2,4), (4,2) и резервными узлами (3,3), (4,1), (4,4) на рис. 5 показана эволюция маршрутов в процессе устранения пересечений. Значения клеточных переменных резервных узлов при выборе направлений маршрутов на этапах 1, 2, 3 и результат реконфигурации МПС представлены на рис. 6.

При устранении столкновений сохраняется связь с максимальным втекающим током. Так, для узла (4,1) (рис. 6, *а*) сохраняется связь с направления 1 ($I_1^{4,1} = -50$) и разрывается с направления 3 ($I_3^{4,1} = -27{,}50 \rightarrow y_3^{4,1} = 0$). Аналогично в узле (3,3) сохраняется связь с направления 3 ($I_3^{3,3} = -20{,}66$) и разрывается с направления 1 ($I_1^{3,3} = -17{,}5 \rightarrow y_1^{3,3} = 0$). Реконфигурация МПС выполнена в соответствии с построенными маршрутами (рис. 6, *б*).

(4,1)	
φ	0
I_1	-50
I_2	-18.94
I_3	-27.50
I_4	0
y	1101

(3,3)	
φ	0
I_1	-17.50
I_2	-29.53
I_3	-20.66
I_4	-26.37
y	0111

(4,4)	
φ	0
I_1	0
I_2	-23.96
I_3	-30.43
I_4	-20.67
y	1111



а) б)

Рис. 6. Значения клеточных переменных (*а*) и результат реконфигурации МПС (*б*)

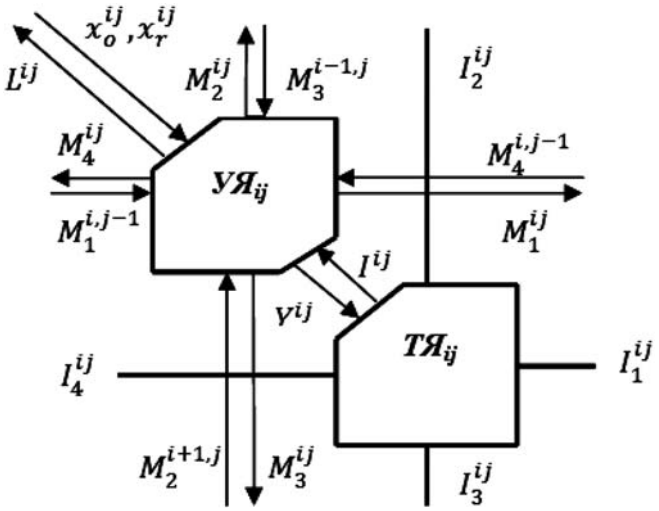


Рис. 7. Структура связей ячейки C_{ij} клеточной среды: УЯ и ТЯ — управляющая и токовая ячейки

Среда самореконфигурации представляет решетку ячеек C_{ij} (см. рис. 1), каждая из которых реализует необходимый набор клеточных операций. При реализации предлагаемой континуальной среды открывается возможность перехода от модели функционирования управляемой токопроводящей решетки (операции Θ_1, Θ_2) к использованию физического аналога решетки, в которой формирование потенциалов и токов происходит по естественным законам.

Использование реальной токопроводящей решетки вместо ее математической модели позволяет исключить операции Θ_1, Θ_2, освобождает от затрат на реализацию вычислений и на хранение значений переменных (I^{ij}), поскольку значения токов хранятся в физических связях решетки.

Структура связей ячейки C_{ij} на базе физического аналога узла токопроводящей решетки для рассмотренного алгоритма формирования маршрутов показана на рис. 7.

Исследование корректирующей способности алгоритма самореконфигурации (относительной доли успешного восстановления к общему числу экспериментов) показало, что при числе отказов, меньшем четырех, корректируются все комбинации отказов для любых размеров решетки независимо от вариантов размещения резерва.

При увеличении числа неисправностей доля фатальных ситуаций для решеток с избыточностью 15—20 % составляет не более 3 %. Так, для решетки 7 × 7 с шестью равномерно размещенными резервными элементами корректирующая способность для предельного числа отказов составила 0,997.

Заключение

Рассмотрен подход к построению отказоустойчивых МПС с решетчатой структурой, основанный на резервировании ПМ соседних элементов в каждом ПЭ и автоматической перенастройке элементов на минимальных маршрутах от мест отказов к резервным узлам. Разработан клеточный алгоритм самореконфигурации отказоустойчивой МПС, использующий модель естественно-подобной среды для параллельного поиска непересекающихся маршрутов восстановления. Клеточные вычислительные среды самореконфигурации на базе управляемой токопроводящей решетки снижают мощность связей ячеек среды и увеличивают корректирующую способность по сравнению с дискретными средами [6].

Универсальная клеточная реконфигурация решетчатых структур отказоустойчивых МПС выполняется автономно на аппаратно-микропрограммном уровне и не зависит от программного и технического обеспечения МПС. Обработка локальных данных об отказах средой реконфигурации обеспечивает масштабирование решетчатых МПС и сохранение способности к реконфигурации при любых размерах решетки и произвольных конфигурациях отказавших и резервных узлов.

Полученные результаты могут быть использованы в гибких масштабируемых многопроцессорных системах с распределенными универсальными средствами реконфигурации, обеспечивающими автономное восстановление функций без перезагрузки системы.

Список литературы

1. **Каравай М. Ф.** Минимизированное вложение произвольных гамильтоновых графов в отказоустойчивый граф и реконфигурация при отказах. II. Решетки и k-отказоустойчивость // Автоматика и телемеханика. 2005. № 2. С. 175—189.

2. **Хорошевский В. Г.** Распределенные вычислительные системы с программируемой структурой // Вестник Сиб. ГУТИ. 2010. № 2. С. 3—11.

3. **Катаев О. В.** Об одном подходе к построению отказоустойчивых бортовых многопроцессорных вычислительных управляющих систем // Искусственный интеллект. 2008. № 4. С. 538—544.

4. **Koloskov V. A., Medvedeva M. V.** Algorithms of Recustomizing of Fault-Tolerant Multicontrollers // Supplement of the 2001 IEEE International Conference on Dependable Systems and Networks. Göteborg. Sweden. July 2001. P. B-22.

5. **Koloskov V. A., Medvedeva M. V.** Models of Active Environment of Self-Organization of Fault-Tolerant Multicontrollers // Programming and Computer Software. 2001. Vol. 27, N 6. P. 67—76.

6. **Колосков В. А., Медведев А. В., Медведева М. В.** Построение клеточных алгоритмов самоорганизации мультимикроконтроллеров с программируемым резервом // Автоматика и телемеханика. 2002. № 1. С. 161—172.

7. **Medvedeva M. V., Koloskov V. A.** Self-Organization of Cellular Environment and Reproduction of the Network Logical Structure // Nuclear Inst. And Methods in Physics Research, A. April 2003. Vol. 502/2-3. P. 540—542.

8. **Колосков В. А., Римский Д. С.** Использование континуально-логических клеточных автоматов для управления реконфигурацией однородной мультиконтроллерной системы // Матер. XVII Всероссийского семинара "Нейроинформатика, ее приложения и анализ данных". Красноярск: ИПК СФУ, 2009. С. 67—69.

УДК 004.272.43

А. Э. Саак, канд. техн. наук, доц.,
Технологический институт Южного федерального университета в г. Таганроге,
e-mail: saak@tti.sfedu.ru

Сравнительный анализ полиномиальных алгоритмов диспетчеризации в Grid-системах

Рассматривается круговой тип массива заявок пользователей на компьютерное обслуживание в Grid-системах, многопроцессорных вычислительных системах. Предлагаются и исследуются уровневый и балансный полиномиальные алгоритмы назначения заявок кругового квадратичного типа. Ранее автором приводились вершинно-кольцевой и однородный полиномиальные алгоритмы диспетчеризации. Проводится сравнительный анализ указанных полиномиальных алгоритмов распределения вычислительных ресурсов и даются рекомендации о возможности их использования в диспетчере как МВС, так и центра Grid-технологий.

Ключевые слова: *Grid- система, многопроцессорная вычислительная система, диспетчирование, круговой квадратичный тип массива требований пользователей, уровневый полиномиальный алгоритм, балансный полиномиальный алгоритм*

1. Постановка задачи

В работах [1, 2] для множества заявок пользователей на компьютерное обслуживание в Grid-системах, многопроцессорных вычислительных системах (МВС) [4—8] построена квадратичная классификация на круговые, гиперболические и параболические массивы. Для требований кругового квадратичного типа в работе [2] описан вершинно-кольцевой, в работе [3] приведен однородный алгоритм диспетчеризации. В настоящей статье предлагаются и исследуются уровневый и балансный алгоритмы назначения на обслуживание заявок кругового квадратичного типа. Ставится задача провести сравнительный анализ указанных полиномиальных алгоритмов распределения ресурсов.

2. Уровневый алгоритм

При представлении заявки пользователя для обслуживания диспетчером центра Grid-технологий или операционной системы МВС координатным ресурсным прямоугольником горизонтальное и вертикальное измерения принимаются равными числу единиц ресурса процессоров и времени, требуемому для обработки, соответственно. Символом $a(j_1) \times b(j_1)$ или $[(a(j_1), b(j_1))]$ обозначается j_1-ая заявка, требующая $a(j_1)$ единиц процессоров и $b(j_1)$ единиц времени.

Приведем и исследуем уровневый алгоритм диспетчеризации линейными круговыми полиэдралями координатных ресурсных прямоугольников $\bigcup_{j_1=0}^{k-1}[a(j_1), b(j_1)]$ (рис. 1), $a(j_1 + 1) \leqslant a(j_1)$, $b(j_1 + 1) \leqslant b(j_1)$, $a(j_1) \geqslant b(j_1)$.

Находим уровень $H = \sqrt{\sum_{j_1=0}^{k-1} a(j_1)b(j_1)}$ горизонтальной полосы $0 \leqslant Y_2 \leqslant H$.

На первом шаге вдоль линии $Y_1 = 0$ вертикально суперпозируются ресурсные прямоугольники $\bigcup_{j_1=0}^{q_1-1}[(a(j_1), b(j_1))]$ до наилучшего приближения уровня с недостатком $\sum_{j_1=0}^{q_1-1} b(j_1) = H - 0$, где q_1 — мощность ресурсных прямоугольников, суперпозированных в первом слое. Строится начальная ресурсная оболочка полученной аддитивной графики (рис. 2).

На втором шаге вдоль правой стороны достигнутой ресурсной оболочки $Y_1 = a(0)$ вертикально суперпозируются последующие ресурсные прямоугольники $\bigcup_{j_1=q_1}^{q_2-1}[(a(j_1), b(j_1))]$ до наилучшего приближения уровня с недостатком $\sum_{j_1=q_1}^{q_2-1} b(j_1) = H - 0$,

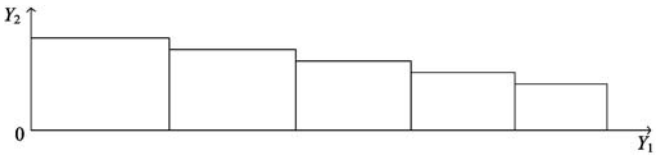


Рис. 1. Круговая линейная полиэдраль ресурсных прямоугольников

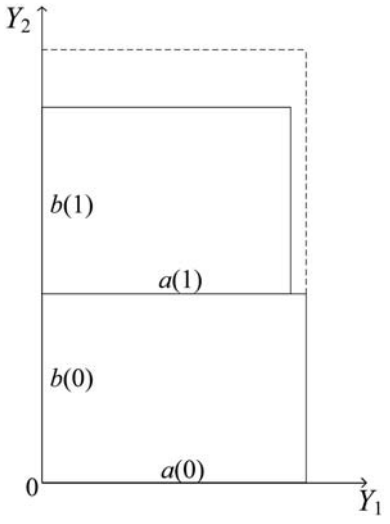


Рис. 2. Начальная ресурсная оболочка

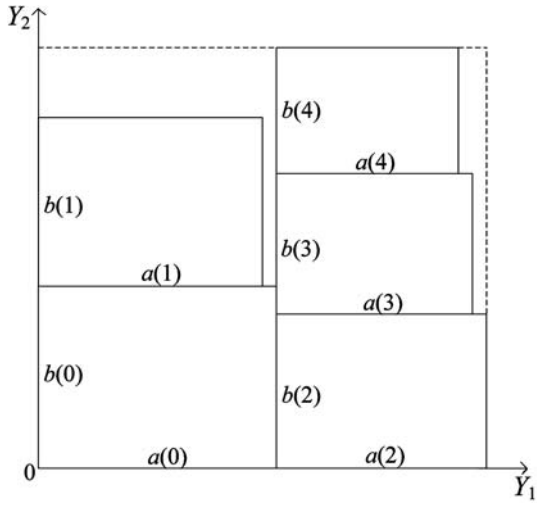


Рис. 3. Первая ресурсная оболочка

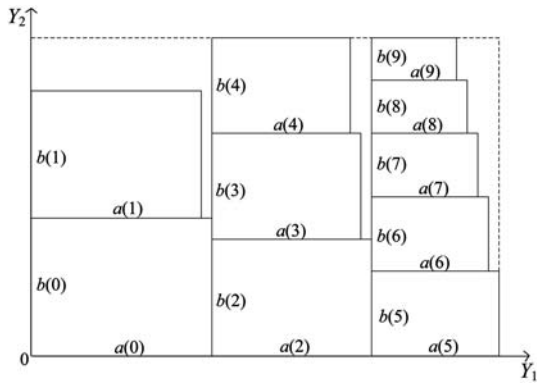


Рис. 4. Конечная ресурсная оболочка

где q_2 — мощность ресурсных прямоугольников, суперпозированных во втором слое. Строится новая ресурсная оболочка (рис. 3).

На следующем шаге вдоль правой стороны достигнутой ресурсной оболочки $Y_1 = a(0) + a(q_1)$ вертикально суперпозируются последующие ресурсные прямоугольники $\bigcup_{j_1 = q_2}^{q_3 - 1} [(a(j_1), b(j_1))]$ до наилучшего приближения уровня с недостатком $\sum_{j_1 = q_2}^{q_3 - 1} b(j_1) = H - 0$, где q_3 — мощность ресурсных прямоугольников, суперпозированных в третьем слое. Строится конечная ресурсная оболочка (рис. 4).

Введенный таким образом уровневый алгоритм повторяем до полного исчерпания ресурсных прямоугольников массива.

Приведем структурную схему уровневого алгоритма (рис. 5). Введем следующие обозначения: $a(j)$, $b(j)$ — параметры требований пользователей, k — число заявок пользователей; $RA(j)$, $RB(j)$ — номера процессорного и временного ресурсов, начиная с которых будет выделено $a(j)$, $b(j)$ единиц соответствующего ресурса для j-й заявки; AL — номер ресурса 1-го рода по оси Y_1, одинаковый для всех прямоугольников соответствующей вертикальной полиэдрали; $H(V)$ — уровень V-й вертикальной полиэдрали; A, $\max H(V)$ — параметры текущей оболочки соответственно по ресурсам 1-го, 2-го рода.

Отметим, что число операций работы уровневого алгоритма составляет k операций сложения и k операций сравнения, т. е. алгоритм является полиномиальным.

В частности, для линейной полиэдрали натуральных ресурсных квадратов при $k = 32$ соответствующие построения приведены на рис. 6.

Время работы оптимального алгоритма [12] приведено в табл. 1.

Сравним качество приведенного уровневого алгоритма с оптимальной укладкой в объемлющий

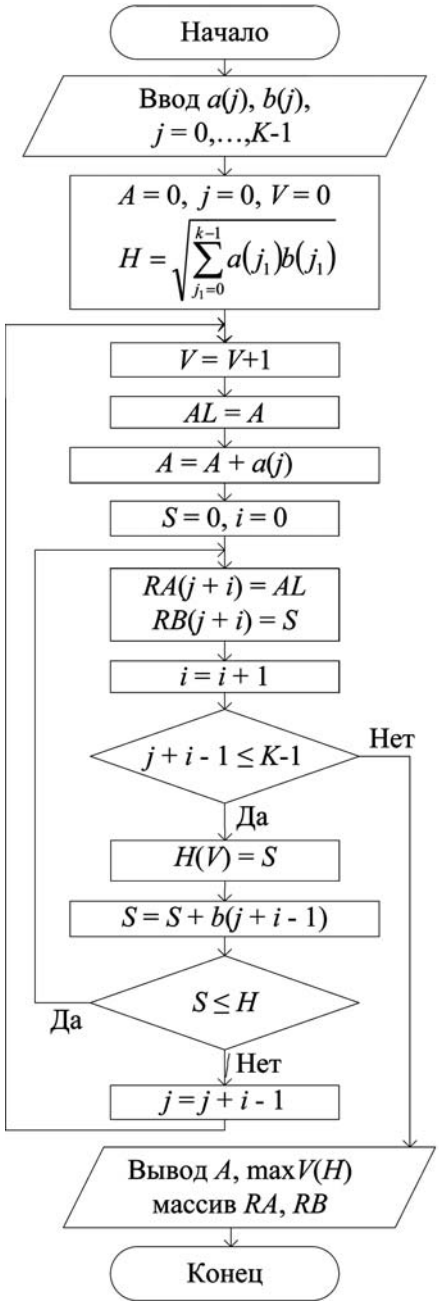


Рис. 5. Структурная схема уровневого алгоритма

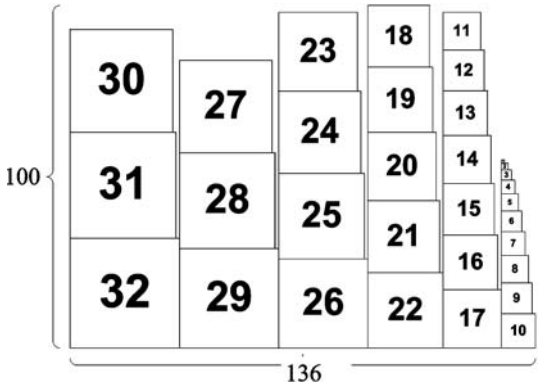


Рис. 6. Укладка линейной полиэдрали ресурсных квадратов уровневым алгоритмом

Таблица 1

Время работы оптимального алгоритма Linux, 2GHz AMD Opteron 246, 2GB RAM

k	Дни	Часы	Минуты	Секунды
27	—	—	35	12
28	—	4	39	31
29	—	8	06	03
30	2	17	32	52
31	4	16	03	42
32	33	11	36	23

Таблица 2

Сравнение уровневого и оптимального алгоритмов

k	$L \times H$	Δ, %	k	$L \times H$	Δ, %
1	1 × 1	0	17	52 × 42	21,7
2	2 × 3	0	18	56 × 46	20,4
3	5 × 3	0	19	63 × 50	26,5
4	7 × 6	20,0	20	69 × 54	28,9
5	11 × 7	28,3	21	74 × 58	28,3
6	14 × 9	27,3	22	80 × 62	29,8
7	17 × 11	21,4	23	80 × 66	21,3
8	17 × 15	21,4	24	85 × 69	19,0
9	21 × 17	19,0	25	90 × 75	21,7
10	25 × 20	23,5	26	96 × 75	15,6
11	30 × 22	28,7	27	101 × 78	13,3
12	34 × 26	32,5	28	108 × 88	22,6
13	39 × 25	16,6	29	116 × 90	21,6
14	43 × 28	16,3	30	123 × 92	19,3
15	44 × 36	25,2	31	129 × 100	28,9
16	48 × 39	23,8	32	136 × 100	18,5

прямоугольник минимальной площади, полученной в работах [9—12]. Приведем в табл. 2 результаты размещения уровневым алгоритмом для последовательности натуральных ресурсных квадратов от 1×1 до $k \times k$. Здесь L — горизонтальное и H — вертикальное измерения объемлющего прямоугольника уровневого алгоритма; Δ — погрешность площади ресурсной оболочки в процентах относительно оптимального значения.

Видим, что погрешность не превосходит 33 %, что является подтверждением целесообразности использования предложенного уровневого алгоритма при диспетчировании процессорно-временными ресурсами.

3. Балансный алгоритм

Приведем и исследуем балансный алгоритм диспетчеризации линейными круговыми полиэдралями координатных ресурсных прямоугольников $\bigcup_{j_1=0}^{k-1}[a(j_1),\ b(j_1)]$.

В общем случае, балансный алгоритм основан на соотношении равенства суммарных уровней

$$\sum_{j_1=0}^{q-1} b(j_1) = \sum_{j_1=q}^{k-1} b(j_1) \pm 0$$

двух вертикальных суперпозиций

$$\bigcup_{j_1=0}^{q-1}[(a(j_1),\ b(j_1))],\ \bigcup_{j_1=q}^{k-1}[(a(j_1),\ b(j_1))]$$

координатных ресурсных прямоугольников линейной полиэдрали (рис. 7).

Координатная ресурсная оболочка балансного алгоритма такова: $(a(0) + a(q)) \times H(q)$. Здесь $H(q)$ — превалирующий уровень левой и правой полиэдралей (рис. 7).

Так, для линейной полиэдрали натуральных ресурсных квадратов $\bigcup_{j_1=0}^{k-1}(k-j_1) \times (k-j_1)$ уравнение баланса принимает следующий вид:

$$H(q) = \sum_{j_1=0}^{q-1}(k-j_1) = \sum_{j_1=q}^{k-1}(k-j_1),$$

или

$$\sum_{j_1=k-q+1}^{k} j_1 = \sum_{j_1=1}^{k-q} j_1.$$

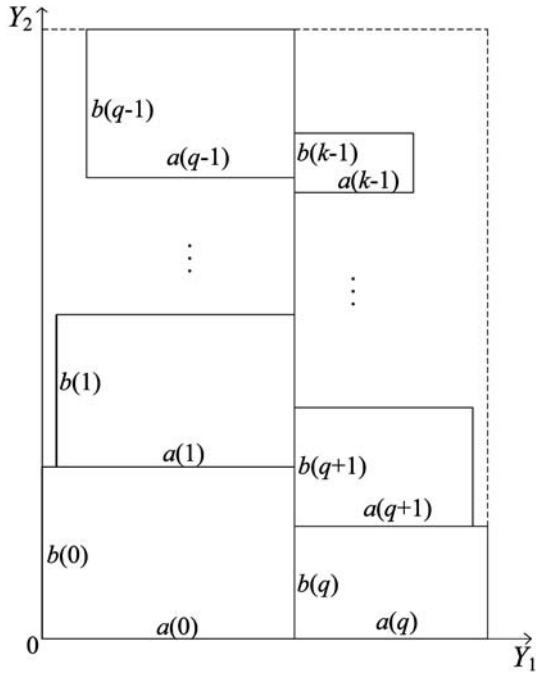


Рис. 7. Балансный алгоритм упорядочивания

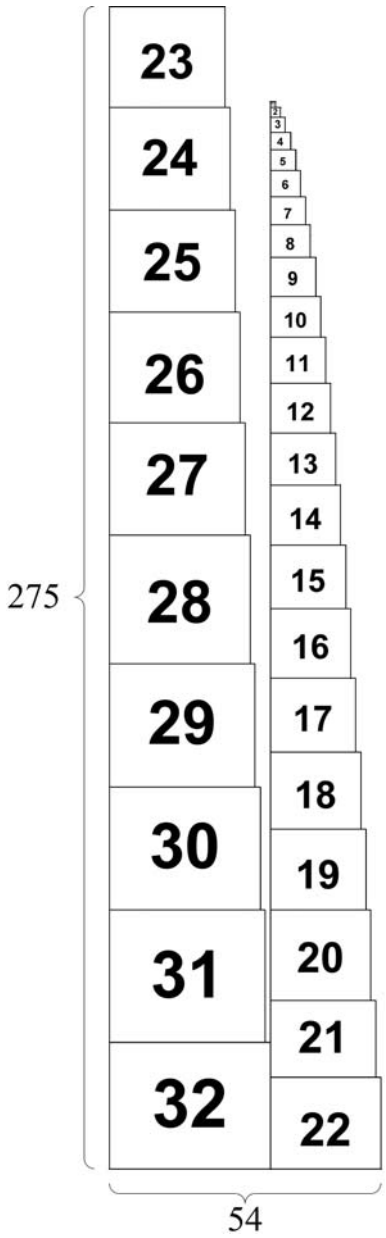


Рис. 8. Укладка линейной полиэдрали ресурсных квадратов балансным алгоритмом

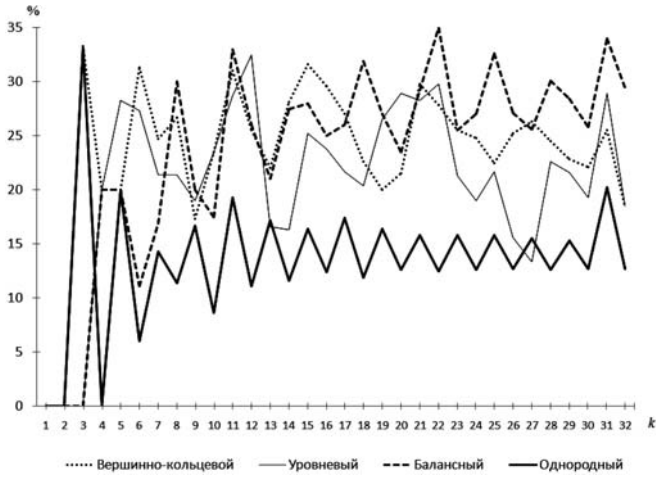


Рис. 9. Сравнение полиномиальных алгоритмов диспетчеризации

Высота каждой полиэдрали должна быть равна половине суммы высот всех ресурсных квадратов:

$$(k - q)(k - q + 1) = \frac{k(k+1)}{2}.$$

Решив квадратное уравнение, получим

$$q^2 - (2k + 1)q + \frac{k(k+1)}{2} = 0,$$

$$q_{1,\,2} = \frac{(2k+1) \pm \sqrt{(2k+1)^2 - 4\dfrac{k(k+1)}{2}}}{2} =$$

$$= \frac{(2k+1) \pm \sqrt{(k+1)^2 + k^2}}{2}.$$

В частности, при $k = 32$ имеем $q_{1,2} = \dfrac{65 \pm 45}{2}$, выбираем $q = 10$. Соответствующие построения приведены на рис. 8.

Проведем сравнительный анализ разработанных и исследованных полиномиальных алгоритмов диспетчеризации линейными полиэдралями кругового типа. Графики погрешностей по сравнению с оптимальным алгоритмом приведены на рис. 9.

Расчеты показали, что погрешность вершинно-кольцевого алгоритма не превосходит 35 %, погрешность балансного алгоритма не превосходит 35 %, погрешность уровневого алгоритма не превосходит 33 %, погрешность однородного алгоритма не превосходит 21 %. Таким образом, все предложенные полиномиальные алгоритмы распределения процессорно-временными ресурсами могут быть рекомендованы для использования в диспетчерах как МВС, так и центра Grid-технологий.

Заключение

В статье для кругового типа массива заявок пользователей рассмотрены уровневый и балансный полиномиальные алгоритмы диспетчеризации. Проведен сравнительный анализ уровневого, балансного, вершинно-кольцевого и однородного полиномиальных алгоритмов распределения вычислительных ресурсов и даны рекомендации о возможности их использования в диспетчере как МВС, так и центра Grid-технологий.

Список литературы

1. **Саак А. Э.** Локально-оптимальные ресурсные распределения // Информационные технологии. 2011. № 2. С. 28—34.

2. **Саак А. Э.** Алгоритмы диспетчеризации в Grid-системах на основе квадратичной типизации массивов заявок // Информационные технологии. 2011. № 11. С. 9—13.

3. **Саак А. Э.** Диспетчеризация в Grid-системах на основе однородной квадратичной типизации массивов заявок пользователей // Информационные технологии. 2012. № 4. С. 32—36.

4. **Барский А. Б.** Параллельные информационные технологии. М.: ИНТУИТ; БИНОМ. Лаборатория знаний, 2007. 503 с.

5. **Барский А. Б.** Параллельные информационные технологии в основе Grid-системы // Информационные технологии. 2006. № 12. С. 54—60.

6. **Хорошевский В. Г.** Архитектура вычислительных систем. М.: Изд-во МГТУ им. Н. Э. Баумана, 2005. 512 с.

7. **Воеводин В. В., Воеводин Вл. В.** Параллельные вычисления. СПб.: БХВ-Петербург, 2002. 608 с.
8. **Каляев И. А., Левин И. И., Семерников Е. А., Шмойлов В. И.** Реконфигурируемые мультиконвейерные вычислительные структуры. Изд. 2-е, перераб. и доп. / Под общ. ред. И. А. Каляева. Ростов н/Д: Изд-во ЮНЦ РАН, 2009. 344 с.
9. **Korf R.** Optimal rectangle packing: Initial results // Proc. of the thirteenth international conference on automated planning and scheduling (ICAPS 2003). Trento: AAAI Press, 2003. P. 287—295.
10. **Korf R.** Optimal rectangle packing: New results // Proc. of the fourteenth international conference on automated planning and scheduling (ICAPS 2004). Whistler: AAAI Press, 2004. P. 142—149.
11. **Korf R., Moffitt M., Pollack M.** Optimal rectangle packing // Annals of Operations Research. 2010. Vol. 179. N 1. P. 261—295.
12. **Korf R., Huang E.** (2009). New Improvements in Optimal Rectangle Packing // Proc. of the 21st International Joint Conference on Artificial Intelligence (IJCAI 2009). Pasadena, California, USA, July 11—17, 2009. P. 511—516.

УДК 621.31

С. В. Дворников[1], д-р техн. наук, доц., проф. каф.,
e-mail: practicdsv@yandex.ru
Е. В. Казаков[2], нач. лаб.,
А. А. Устинов[1], д-р техн. наук, проф.,
зам. начальника каф.,
А. П. Чихонадских[2], канд. техн. наук,
ст. науч. сотр., начальник науч. центра,
С. В. Андриянов[1], курсант
[1] Военная академия связи
[2] ГосНИИПП

Обоснование модели секвентного сигнала для систем связи

Предлагаются результаты аналитических исследований и данные компьютерного эксперимента по обоснованию выбора математической модели сигнала без несущей для его применения в системах связи при передаче информации. Обосновывается выбор двуполярного сигнала на основе импульсов Гаусса с минимальным сдвигом между их медианными значениями.

Ключевые слова: *секвентные сигналы, аналитическая модель сигнала, спектральная эффективность, функция Гаусса*

Введение

Для борьбы с преднамеренными помехами с середины XX века активно ведутся работы по созданию систем связи, использующих широкополосные сигналы (*broadband signal* — BBS) или "шумоподобные сигналы" — (*noise-like signal*), обеспечивающие высокую энергетическую скрытность по отношению к комплексам радиомониторинга.

Как правило, формирование шумоподобных сигналов связано с технологий расширения спектра. Термин "расширение спектра" обусловлен тем, что полоса частот, используемая для передачи сигнала, намного шире минимальной, необходимой для передачи данных [1]. Между тем, свойство широкополосности присуще и так называемым секвентным сверхкратковременным импульсам (ССИ) нано- и пикосекундных длительностей [2], для которых в 1992 г. Управлением перспективных исследовательских программ министерства обороны США (*Defense Advanced Research Projects Agency* — DARPA) введено понятие "сверхширокополосные сигналы" (*Ultra-wide band* — UWB).

Теоретической предпосылкой применения ССИ служит теорема Шеннона, в соответствии с которой количество информации, передаваемое по каналу связи, зависит от ширины полосы частот, занимаемой сигналом, увеличение которой возможно за счет уменьшения его длительности. Впервые наиболее полно данный вопрос был рассмотрен в работе [2] Х. Хармутом, где указанные сигналы получили название секвентных, т. е. сигналов без несущей. К основным достоинствам систем связи, использующих ССИ, следует отнести:

- высокую скорость передачи данных (до сотен мегабит/с);
- защищенность от активных узкополосных и широкополосных помех;
- возможность использования каналов с многолучевым распространением радиоволн;
- низкую спектральную плотность средней излучаемой мощности.

В работе [3] представлены математические описания моделей ССИ, которые потенциально можно использовать в интересах передачи информации. Однако до сих пор нет четкой концепции их применения [4]. Между тем, данный вопрос требует серьезного исследования в целях выработки общих принципов и подходов. Настоящая статья затрагивает лишь один из аспектов рассматриваемой проблематики, связанный с обоснованием модели секвентного сигнала для системы связи, с позиции его спектральной эффективности.

1. Описание секвентных сверхкратковременных импульсов

Основным показателем, характеризующим свойство широкополосности сигнала, является его база, рассчитываемая в соответствии с формулой

$$B = \Delta F T_c,$$

где T_c — длительность сигнала; ΔF — занимаемая полоса частот.

Сигнал считается широкополосным, если его база больше единицы

$$B > 1. \quad (1)$$

Как правило, известные технологии расширения спектра базируются на применении частотно-временных матриц при формировании сигнальных конструкций. Между тем секвентные сигналы нано и пикосекундной длительности, согласно (1) не являются широкополосными, поскольку формально их база $B = 1$. При этом полоса диапазона занимаемых ими частот в ряде случаев существенно шире, чем у сигналов, база которых значительно больше единицы [1]. В связи с этим, для определения широкополосности целесообразнее использовать тезаурус, предложенный DARPA, согласно которому данное понятие рассматривается с позиций относительного значения полосы частот, занимаемой сигнальной выборкой

$$\widehat{B} = \frac{f_{\max} - f_{\min}}{f_{\max} + f_{\min}}, \quad (2)$$

где $f_{\max}$, $f_{\min}$ — соответственно верхняя и нижняя граница действующей полосы частот сигнала.

Тогда при условии $\widehat{B} \leqslant 0{,}1$ сигнал считается узкополосным, при $0{,}1 < \widehat{B} \leqslant 0{,}25$ — широкополосным, а при $0{,}25 < \widehat{B} \leqslant 1$ — сверхширокополосным.

С позиций выражения (2) ССИ можно отнести к сверхширокополосным сигналам, что в полной мере отражает их физическую сущность.

В целях дальнейшего упорядочения представления материала введем следующие понятия и определения (рис. 1).

Входная реализация $E(t)$ — совокупность отсчетов, поступающих с аналого-цифрового преобразователя.

Выборка $z(t)$ — совокупность отсчетов входной реализации, представленных для обработки.

Полезный сигнал $s(t)$ — электромагнитное колебание (секвентный сигнал), выделяемое из совокупности радиоизлучений, содержащихся в выборке.

Слот — выборка τ_0, равная длительности полезного сигнала $s(t)$.

Фрейм — выборка T_0, содержащая совокупность слотов t_0, в пределах одного из которых всегда содержится полезный сигнал $s(t)$, при условии его излучения.

Сигнальная выборка — выборка T_c, содержащая совокупность фреймов T_0, в каждом из которых имеется полезный сигнал $s(t)$ в пределах одного из слотов, и характеризующая информационный символ $S(t)$ логического нуля или единицы (число фреймов, используемых для передачи одного бита).

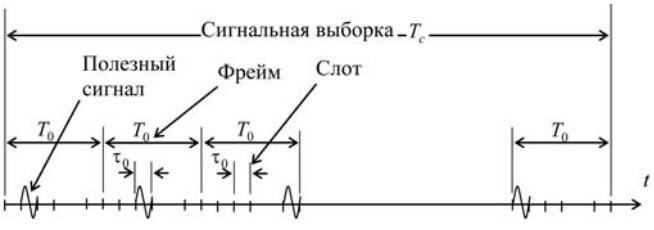


Рис. 1. Структура сигнальной выборки на основе ССИ

Согласно работе [5], кодирование информации сигнальной выборки на основе ССИ осуществляется с использованием время-импульсной модуляции (ВИМ), в соответствии с которой значение логической единицы или нуля определяется временным положением полезного сигнала в пределах фрейма.

По своей сущности ВИМ предполагает различные подходы к информационному кодированию сигналов логических единиц и нулей. Рассмотрим некоторые из возможных вариантов.

В условиях, когда сигнальная выборка равна длительности фрейма

$$T_c = T_0$$

простейший способ информационного кодирования реализуется путем излучения (не излучения) полезного сигнала в пределах фрейма. Его достоинство в простоте подхода, не требующего высокостабильной тактовой синхронизации. Однако при наличии серий повторяющихся логических нулей (при условии нуль — отсутствие излучения) возможен срыв синхронизации.

Другой подход к информационному кодированию заключается в жестком определении позиций слотов, в пределах которых будут излучаться или логические единицы, или логические нули. Такой подход требует четкой синхронизации по слотам. Частным случаем его реализации является закрепление первой половины слотов фрейма для передачи, например, логической единицы, а второй — логических нулей. В этом случае синхронизацию достаточно осуществлять с точностью до $T_0/2$.

Интересным видится информационное кодирование на основе нескольких полезных сигналов, в этом случае

$$T_c = NT_0,$$

где N — число фреймов, определяющих логический сигнал (один бит).

Указанный подход, предусматривающий наличие правила, согласно которому происходит отбор фреймов для формирования информационного бита, обеспечивает определенную структурную скрытность передаваемым сообщениям.

Таким образом, ВИМ является эффективным инструментом информационного кодирования ССИ.

2. Выбор и обоснование формы полезного сигнала

Рассмотренный принцип информационного кодирования в системах связи, использующих ССИ, позволяет заключить, что важным моментом, определяющим характер их работы, является выбор формы полезного сигнала.

С одной стороны, желательно, чтобы ССИ имел относительно простое аналитическое описание, позволяющее реализацию быстрых алгоритмов его синтеза. С другой стороны, он должен обладать ря-

дом положительных свойств, делающих его применение эффективным с точки зрения выбранной целевой установки. Примерами таких свойств являются однородность плотности спектральной энергии в пределах заданной полосы, неразрывность фазовых характеристик, максимальная концентрация спектральной мощности в пределах главного "лепестка" и др. Аналитическая функция полезного сигнала $s(t)$ в виде ССИ всегда равна нулю вне слота:

$$z(t) = \begin{cases} s(t), \text{ если } t = \tau_0; \\ 0, \text{ если } t \neq \tau_0. \end{cases} \quad (3)$$

Спектральное представление (3) описывается комплексной функцией

$$\dot{A}(f) = \int_{-\infty}^{+\infty} z(t)\exp(-j2\pi ft)dt = \int_{-\tau_0/2}^{\tau_0/2} s(t)\exp(-j2\pi ft)dt. \quad (4)$$

Для случая, когда τ_0 достаточно мало, значение $\exp(\pm j2\pi f\tau_0/2)$ близко к единице и решение (4) имеет вид

$$\int_{-\tau_0/2}^{\tau_0/2} s(t)dt = q,$$

где q — константа, определяемая энергетическими параметрами $s(t)$.

В работе [3] обосновано, что для рассматриваемого случая близость значения $\exp(\pm j\pi f\tau_0)$ к единице достигается только при $f\pi\tau_0 \ll 1$, что соответствует условию $\tau_0 \ll T_0$, где T_0 — период, соответствующий частоте $f(T_0 - 1/f)$.

Анализ полученных аналитических результатов показывает, что одиночный импульс независимо от своей формы всегда имеет сплошной спектр, энергия которого пропорциональна энергии импульса в пределах того интервала частот, в котором его период остается относительно большим по сравнению с длительностью самого импульса.

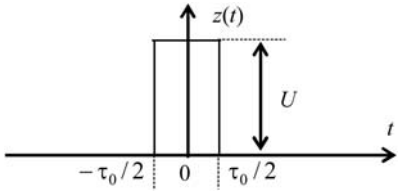


Рис. 2. Временное представление прямоугольного импульса

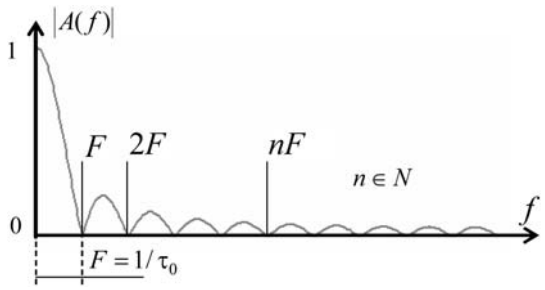


Рис. 3. Спектр (распределение плотности спектральной энергии) прямоугольного импульса

Однако с повышением частоты, когда значение T_0 практически сравнимо со значением τ_0, функция (4) асимптотически убывает.

Простейшую аналитическую модель имеет прямоугольный импульс

$$z(t) = \begin{cases} U, \text{ если } t \in [-\tau_0/2;\ 0]; \\ U, \text{ если } t \in [0;\ \tau_0/2]; \\ 0, \text{ если } t \notin [-\tau_0/2;\ \tau_0/2], \end{cases} \quad (5)$$

где U — амплитудное значение.

Общий вид импульса прямоугольной формы показан на рис. 2.

Для расчета спектра импульса (5) подставим его значение в выражение (4):

$$\dot{A}(f) = \int_{-\infty}^{+\infty} U\exp(-j2\pi ft)dt = \int_{-\tau_0/2}^{\tau_0/2} U\exp(-j2\pi ft)dt =$$
$$= U\int_{-\tau_0/2}^{\tau_0/2} (\cos(2\pi ft) - j\sin(2\pi ft)dt =$$
$$= 2U\int_{0}^{\tau_0/2} (\cos(2\pi ft)dt = \frac{U}{\pi f}\sin(\pi f\tau_0).$$

Сделав замену переменной $x = \pi f\tau_0$, получим $\dot{A}(x) = U\tau_0\frac{\sin(x)}{x}$. Для импульса единичной амплитуды $U = 1$ имеем $\dot{A}(x) = \tau_0\frac{\sin(x)}{x}$.

На практике удобней работать с модулем спектральной плотности (амплитудным спектром) $|A(f)|$, характеризующим распределение энергии импульса вдоль частотной оси. На рис. 3 представлена нормированная спектральная функция прямоугольного импульса (5).

Поскольку для систем связи, использующих ССИ, особый интерес представляют гладкие излучения, имеющие более равномерное распределение энергии в частотном диапазоне, то при выборе формы полезного сигнала будем руководствоваться параметрами одноперiодного синусоидального импульса, центрированного относительно оси ординат (рис. 4). Его аналитическое описание имеет следующий вид:

$$z(t) = \begin{cases} U\sin(2\pi t/\tau_0), \text{ если } t \in [-\tau_0/2;\ 0]; \\ -U\sin(2\pi t/\tau_0), \text{ если } t \in [0;\ \tau_0/2]; \\ 0, \text{ если } t \notin [-\tau_0/2;\ \tau_0/2]. \end{cases} \quad (6)$$

Спектр импульса (6) рассчитаем в соответствии с выражением (4):

$$\dot{A}(f) = \int_{-\tau_0/2}^{\tau_0/2} z(t)\exp(-j2\pi ft)dt = \int_{-\tau_0/2}^{0} z(t)\exp(-j2\pi ft)dt +$$
$$+ \int_{0}^{\tau_0/2} z(t)\exp(-j2\pi ft)dt = -2j\int_{0}^{\tau_0/2} \sin(2\pi ft/\tau_0)\sin(2\pi ft)dt. \quad (7)$$

Интегрируя (7) по частям, получим

$$\dot{A}(f) = j4\pi\tau_0 \frac{\sin(2\pi ft/\tau_0)}{(4\pi^2 f^2 \tau_0^2 - 4\pi^2)} = j\tau \frac{\sin(2\pi ft/\tau_0)}{\pi(f^2\tau_0^2 - 1)}.$$

На рис. 5 показан спектр синусоидального импульса на фоне спектра прямоугольного импульса.

Анализ полученных результатов позволяет заключить, что спектр синусоидального импульса имеет большую концентрацию спектральной энергии в районе сосредоточения ее максимума. Между тем в работе [3] обоснована спектральная эффективность полезных сигналов на основе функции Гаусса, в общем случае аналитическое представление которой, применительно к рассматриваемой проблематике, можно записать в следующем виде:

$$s(t) = \exp(-\alpha t^2). \qquad (8)$$

В выражении (8) α — коэффициент масштабирования. Учитывая, что эффективная длительность Гауссова импульса определяется из условия десятикратного уменьшения его мгновенного значения [3], рассчитаем область допустимых значений α, определив

$$0{,}1 = \exp(-\alpha(t/2)^2). \qquad (9)$$

Потенцируя (9), получим $\ln(0{,}1) = -\alpha(t/2)^2$, откуда можно найти

$$t\sqrt{\alpha} = 2\sqrt{-\ln(0{,}1)} = 3{,}035.$$

В общем случае функция Гаусса безгранична на области определения своего аргумента t, поэтому для синтеза на ее основе двуполярного импульса, ограничим область определения аргумента пределами $[-\tau_0/2;\ \tau_0/2]$. В итоге аналитическая модель для ее описания будет иметь следующий вид:

$$z(t) = \begin{cases} U\exp(-\alpha(t + \tau_0/b)^2, \text{ если } t \in [-\tau_0/2;\ 0]; \\ -U\exp(-\alpha(t - \tau_0/b)^2, \text{ если } t \in [0;\ \tau_0/2]; \\ 0, \text{ если } t \notin [-\tau_0/2;\ \tau_0/2], \end{cases} \qquad (10)$$

где b — множитель, регулирующий медианный разнос формирующих импульсов. В соответствии с функцией (4) при $U = 1$ спектр функции (10) будет

$$\dot{A}(f) = \int_{-\tau_0/2}^{\tau_0/2} z(t)\exp(-j2\pi ft)dt =$$

$$= \int_{-\tau_0/2}^{0} \exp(-\alpha(t + \tau_0/4)^2)\exp(-j2\pi ft)dt -$$

$$- \int_{0}^{\tau_0/2} \exp(-\alpha(t - \tau_0/4)^2)\exp(-j2\pi ft)dt. \qquad (11)$$

Согласно [3], решение уравнения (11) имеет следующий вид:

$$\dot{A}(f) = \sqrt{\pi/\alpha}\left(\exp\left[\frac{(j\alpha\tau_0 - 2\pi f)\pi f}{2\alpha}\right] - \right.$$

$$\left. - \exp\left[\frac{-(j\alpha\tau_0 + 2\pi f)\pi f}{2\alpha}\right]\right). \qquad (12)$$

Анализ спектральной функции (12) показывает, что она определяется отношением f/α и значением τ_0, а также значением сдвига положения формирующих импульсов относительно друг друга на временной оси, максимальный сдвиг которых (рис. 6), обеспечивающий минимальное перекрытие, будет при $\tau_0/4$ (формула (11) приведена для данного случая).

Заметим, что форма огибающей (12) (рис. 7) непосредственно зависит от значения сдвига между положениями максимумов формирующих импульсов на временной оси. Между тем визуальный анализ спектра функции (10) (рис. 7) указывает на наличие в нем ярко выраженных провалов, что является нежелательным явлением. Следовательно, функцию (10) нецелесообразно использовать на практике в качестве модели ССИ. Избежать указанных провалов, обусловленных разрывом формирующей функ-

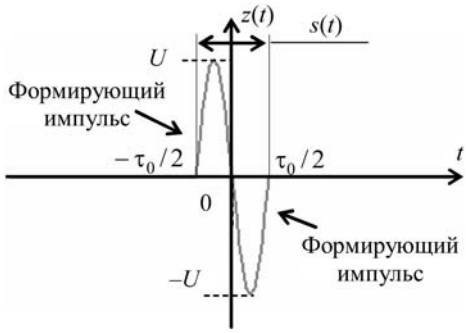


Рис. 4. Временное представление синусоидального импульса

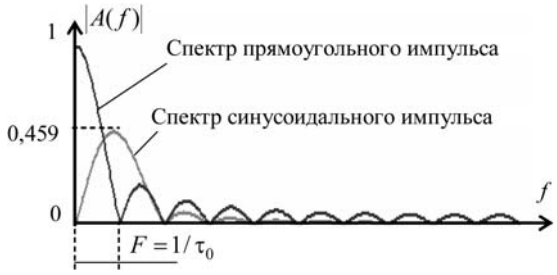


Рис. 5. Спектры синусоидального и прямоугольного импульсов одинаковой длительности

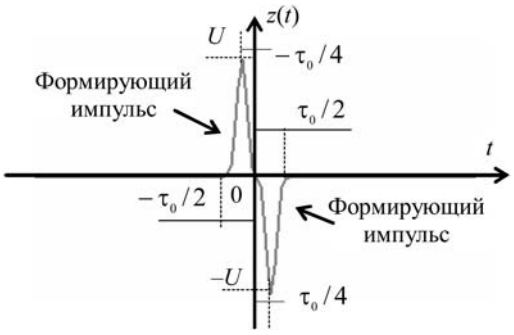


Рис. 6. Временное представление Гауссова импульса (сдвиг максимальный, перекрытие минимальное)

ции в точке ноль [0], возможно за счет повышения гладкости выбранной модели.

Так, на рис. 8 представлен ССИ, сформированный путем максимального перекрытия ($\tau_0/8$) формирующих импульсов.

Поскольку функция, представленная на рис. 8, имеет максимальную степень гладкости для моде-

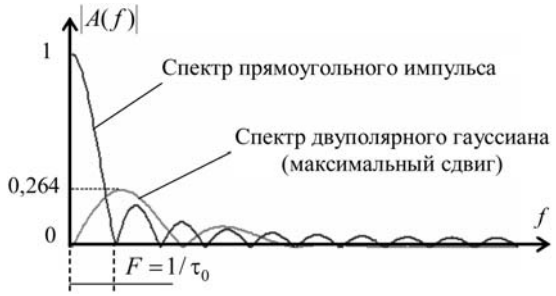


Рис. 7. Спектры прямоугольного и Гауссова импульсов (максимальный сдвиг формирующих импульсов) одинаковой длительности

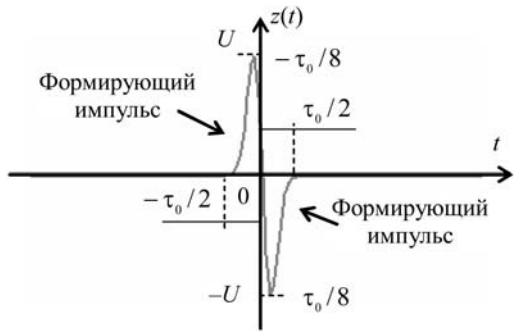


Рис. 8. Временное представление Гауссова импульса (сдвиг минимальный, перекрытие максимальное)

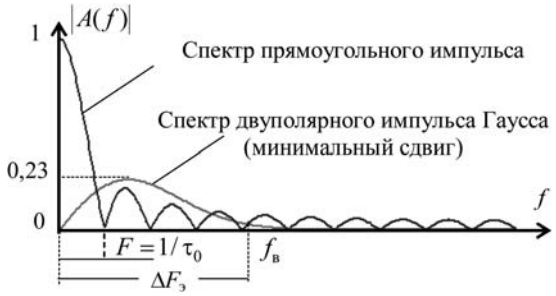


Рис. 9. Спектры прямоугольного импульса и Гауссова импульсов (минимальный сдвиг формирующих импульсов) одинаковой длительности

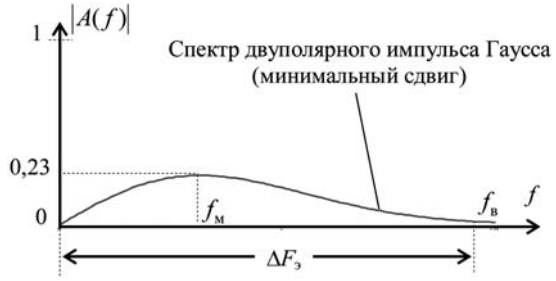


Рис. 10. Рабочая часть спектра двуполярного гауссиана (минимальный сдвиг формирующих функций)

ли (10), то ее спектр (рис. 9) не имеет провалов (энергия сосредоточена в пределах главного "лепестка" ("лепесток" только один)).

Для анализа характера спектральной функции определим значения ее частотных границ $\Delta F_э$ (рис. 10), охватывающих 99 % энергии импульса. Поскольку данная функция является монотонно убывающей справа, то предлагается диапазон $\Delta F_э$ определять по граничному значению $f_в$.

Важным параметром спектральной функции является ее максимальное значение $f_м$. Для его определения необходимо найти производную от (12) по параметру f. Согласно расчетам, максимальное амплитудное значение, соответствующее $f_м$, составляет 0,233 от нормированного значения спектральной функции прямоугольного импульса. Причем максимальная спектральная компонента $f_м$ делит диапазон $\Delta F_э$ в соотношении 1/3, а амплитудное отношение $f_в/f_м$ равно 0,05.

Отметим также, что ширина спектра двуполярного импульса Гаусса при минимальном сдвиге формирующих импульсов в 4,6 раза шире первого "лепестка" спектра прямоугольного импульса той же длительности.

В результате экспериментального исследования зависимости характера спектральной функции от значения сдвига установлено, что максимальный разнос между медианами формирующих импульсов, при котором итоговый спектр еще носит "однолепестковый" характер, не должен превышать $\tau_0/5$.

Заключение

Анализ аналитических расчетов и результатов моделирования показал, что в качестве моделей ССИ целесообразно использовать двуполярные моноимпульсы, полученные на основе формирующих функций Гаусса, взятых с противоположным знаком и имеющих минимально возможный сдвиг между их максимальными значениями. Такой выбор позволит обеспечить наилучшую спектральную эффективность сигналов в частотном диапазоне с точки зрения равномерности распределения энергии.

Дальнейшее направление исследований, по мнению авторов, видится в рассмотрении моделей ССИ на основе производных формирующих функций, а также степенных преобразований.

Список литературы

1. **Григорьев В. А.** Сигналы современных зарубежных систем электросвязи: учебник. СПб.: ВАС, 2007. 368 с.
2. **Хармут Х.** Теория секвентного анализа. Основы и применения. М.: Мир, 1980. 576 с.
3. **Дворников С. В., Железняк В. К.** Основы теории модулированных колебаний. СПб.: ГУАП, 2006. 160 с.
4. **Челышев В. Д., Потапов С. Г.** и др. UWB — Начальные представления во временной и спектральной областях // Информация. Космос. 2007. № 1. С. 45—59.
5. **Имморев И. Я., Судаков А. А.** Сверхширокополосная помехоустойчивая система скрытой связи с высокой скоростью передачи данных // Труды Всероссийской научной конференции семинара "Сверхширокополосные сигналы в радиолокации, связи и акустике" (СРСА — 2003), Муром, июль 2003. С. 435—440.

УДК 004.75: 621.396

А. П. Мощевикин, канд. физ.-мат. наук, доц.,
А. С. Галов, аспирант, программист 1 кат.,
А. С. Волков, программист 1 кат.,
Петрозаводский государственный университет (ПетрГУ)

Точность расчета локации в беспроводных сетях датчиков стандарта nanoLOC (IEEE 802.15.4a)

Рассмотрено применение технологии беспроводных сетей датчиков nanoLOC™ (IEEE 802.15.4a), используемой для построения локальных систем позиционирования мобильных объектов. Описаны принципы расчета локации и даны рекомендации по оценке точности ее расчета.

Ключевые слова: *беспроводные сети датчиков, локальные системы позиционирования объектов, расчет местоположения, nanoLOC™, IEEE 802.15.4a*

Введение

Измерение расстояний между двумя радиоузлами в технологии nanoLOC™ (стандарт IEEE 802.15.4a) существляется с помощью метода Round Trip Time (RTT). Радиоузлы обмениваются кадрами, фиксируя время отправления, обработки и приема кадров. На основании временных меток рассчитывается время распространения сигнала в эфире. На основании вычисленного времени распространения сигнала и константы скорости света рассчитывается расстояние между радиоузлами [1, 2].

В идеальных условиях (при прямолинейном распространении электромагнитной волны и при отсутствии ошибок измерений) измеренное расстояние будет равно действительному. Однако на практике вследствие отражений сигнала результат измерений расстояния будет завышен (сигнал распространяется не по прямой — Non-Line Of Sight (NLOS). Подробно о точности измерения расстояний с помощью трансиверов nanoLOC™ изложено в работе [3].

Задача локации состоит в однозначном определении местоположения мобильного узла относительно стационарных базовых станций (БС). Для случая двумерной геометрии на плоскости это возможно при наличии минимум трех БС. Для такого случая существуют различные методы расчета локаций, например метод триангуляции.

Point-based- и *area-based*-подходы для определения локаций и их точность

Для определения местоположения локации можно пользоваться двумя подходами: *point-based* (определение вероятной точки, в которой находится объект) и *area-based* (определение вероятной области нахождения объекта) [4]. При использовании последнего возможны варианты в способе задания формы области, которая, как правило, характеризуется некоторым значением вероятности нахождения в ней объекта.

Поскольку в случае использовании технологии nanoLOC™ область локации объекта может быть представлена областью пересечения окружностей с центрами в соответствующих БС и радиусами, равными измеренным расстояниям от БС до объекта, то локацию удобно описывать с помощью *area-based*-подхода. Эта область имеет стопроцентную вероятность нахождения в ней объекта, при условии, что все измерения от БС выполнены без ошибок (рис. 1). Применяя дополнительные алгоритмы расчета, внутри полученной области локации можно выделить точку вероятного нахождения объекта, т. е. использовать *point-based*-подход.

Вероятную точку нахождения объекта внутри вероятной области локации можно рассчитать, например, с помощью метода триангуляции (см. точку *2* на рис. 1). Этот способ расчета соответствует простой модели, в которой для любой БС вероятность нахождения объекта в точке, расположенной внутри окружности, соответствующей измерению расстояния до этой БС, тем больше, чем ближе точка к центру окружности. Другой способ расчета *point-based*-локации внутри вероятной области локации может быть основан на более сложной модели или использовании определенных из экспериментов эвристик и получении в качестве результата, например, точки *1* (см. рис. 1).

Оценку точности локации в зависимости от того, какой подход используется для расчета —

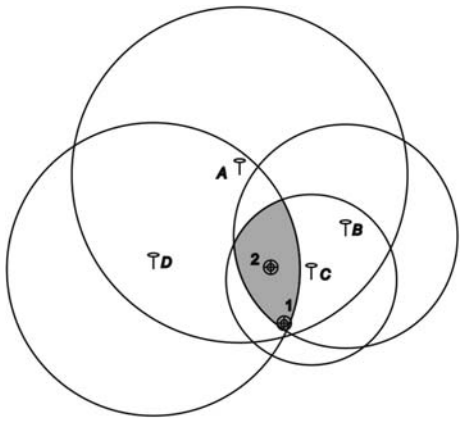


Рис. 1. Область локации объекта

area-based или *point-based*, можно получить по-разному, как:

а) размер зоны вероятного нахождения объекта (вероятность, с которой объект находится внутри зоны, задается заранее);

б) статистическую оценку разброса расчетов локации или кучность;

в) расстояние между действительным положением объекта на местности и рассчитанным.

Первая оценка применяется к *area-based*, а следующие две к *point-based* подходам.

Как обсуждалось выше, измеренные расстояния между базовой станцией и мобильным узлом в технологии nanoLOC^{TM} всегда выше действительных. Это связано с отражениями электромагнитной волны от различных объектов окружения, в том числе от поверхности Земли. Обычно разница между измеренным и действительным значениями расстояний между БС и объектом лежит в пределах 1...10 м, но в определенных случаях она может составлять значительно большие значения [5, 6].

Необходимо отметить, что завышенные результаты измерения расстояний между мобильным объектом и базовой станцией в различной степени влияют на конечную точность определения локации объекта. Большое значение имеет положение мобильного объекта относительно базовых станций. Так, в работе [7] отмечено, что чем больше базовых станций задействовано при измерениях, тем больше вероятность взаимного вычитания ошибок NLOS при вычислении локации (ситуация, когда базовые станции находятся вокруг объекта).

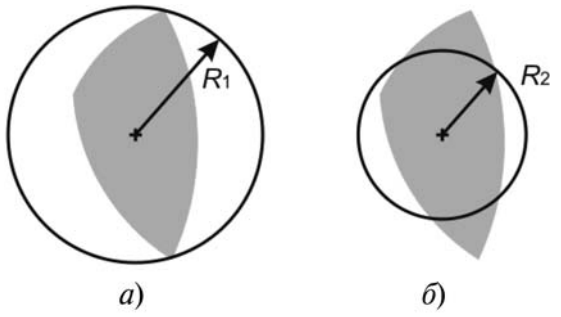


Рис. 2. Возможные оценки области локации

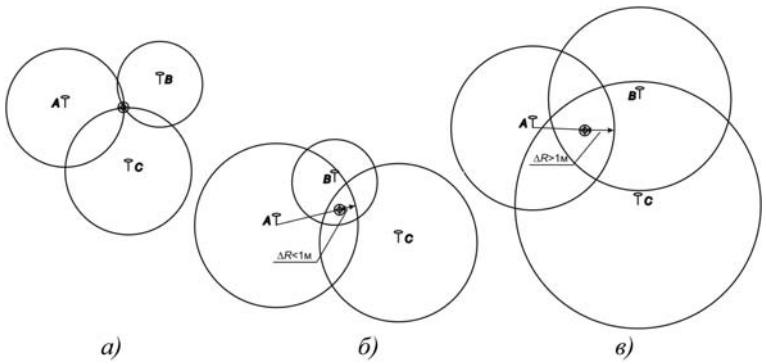


Рис. 3. Различия в относительном размере областей локаций

Если же базовые станции находятся с одной стороны объекта (например, когда объект удаляется от данных базовых станций), то измерения с ошибкой NLOS уменьшают точность расчета локации.

Различные оценки области локации *area-based*-подхода и их точность

В случае *area-based*-подхода важным является выбор простой оценки полученной области локации.

Поскольку формы областей локаций при *area-based*-подходе могут быть различными, то для сравнения двух областей необходима простая оценка этой области, например окружность.

Существует легко реализуемый алгоритм поиска радиуса окружности, описанной вокруг области, сформированной дугами окружностей, соответствующих измерениям от разных базовых станций. Эта область (см. рис. 2, *а*), как было сказано выше, соответствует стопроцентному доверительному интервалу, поэтому недостаток использования описанной окружности в качестве оценки области положения объекта в том, что такая окружность также включает зоны, в которых объект точно находиться не может, — такая оценка является явно завышенной.

Другой способ (рис. 2, *б*) демонстрирует возможность построения окружности такой площади, которая равнялась бы площади области стопроцентного нахождения.

В качестве оценки точности локации для обоих случаев может быть использован радиус соответствующей окружности.

В экспериментах, проводимых коллективом авторов, в качестве оценки точности локации для *area-based*-подхода чаще всего использовалась величина, равная квадратному корню из площади области локации, — это еще один вариант оценки точности *area-based*-локации.

Влияние завышенных результатов измерений на оценку точности локации

Существуют случаи, в которых *area-based*-подход дает в результате расчетов область локации, которая не отражает реальной точности системы. В этом разделе рассматриваются только случаи завышенных результатов измерений, выполненных с ошибкой. Рассмотрим ситуацию, в которой локация определяется по измерениям от трех базовых станций. В идеальном случае, при отсутствии ошибок измерений, все три окружности должны пересекаться в одной точке, — точке действительного положения объекта (см. рис. 3, *а*).

Точность измерения расстояния в отсутствии NLOS, заявленная компанией *Nanotron Technologies GmbH* для технологии nanoLOC^{TM}, составляет 1 м [1, 2]. При условии корректной расстановки базовых стан-

ций на карте, в случае отсутствия NLOS получаем, что расстояние от вычисленной локации до каждой из окружностей не должно превышать 1 м (рис. 3, *б*). Однако при наличии ошибок в измерениях, связанных с отражениями сигналов, пересекающиеся окружности образуют некоторую достаточно большую область (рис. 3, *в*), поскольку полученные результаты измерения значительно больше действительных расстояний между БС и объектом.

Часто в нескольких циклах измерений для определенной БС не все из измерений являются значительно завышенными (под циклом измерений понимается процедура регистрации расстояний между мобильным узлом и несколькими БС). Для одного цикла измерений невозможно определить является ли данное измерение до БС завышенным. Можно определить факт завышения измерения для определенной БС при наличии результатов минимум трех циклов измерений, в двух из которых измерения выполнены без ошибок для соответствующей БС. Оценить степень ошибки NLOS можно по размеру области локации относительно размеров всех окружностей, с помощью которых эта область образована.

Необходимо выделить ситуации, когда пользоваться *area-based*-подходом не следует. Это зависит от взаимного расположения мобильного объекта и БС [8]. Так, в случае, показанном на рис. 4, мобильный объект находится вне треугольника ΔABC, образованного базовыми станциями.

При большом расстоянии от мобильного объекта до базовых станций ошибка, связанная с завышением, редко превышает значение в несколько метров (в противном случае сигнал просто не доходит).

При использовании *area-based*-подхода локация объекта соответствует области пересечения окружностей. Ее площадь может доходить до нескольких сотен и тысяч квадратных метров, что однозначно неприемлемо. Поэтому в ситуации, показанной на рис. 4, в качестве рассчитанной локации объекта правильнее взять точку *1* (точку, близкую к месту пересечения окружностей), а область нахождения объекта определить областью пересечения колец, ограниченных измеренным R и истинным R_0 расстояниями до базовых станций (на рис. 4 для наглядности кольцо с шириной ΔR вырезано только для одной окружности с центром в точке *A*).

Как видно, заштрихованная область явно меньше области, образованной пересечением окружностей. Другими словами область пересечения окружностей не будет отражать реальной точности локации (*area-based*-метрика будет явно завышена).

Обнаружение завышенных результатов в одном цикле измерений

Ошибку, связанную с NLOS, можно легко обнаружить даже в результатах одного цикла измерений при наличии большого числа базовых станций, от которых одновременно измеряются расстояния до исследуемого объекта. В случае, если окружность, связанная с измерением расстояния, не касается области локации объекта, а сама область находится внутри окружности, то измерение, очевидно, является завышенным, и его использование для определения *point-based*-локации на основе области локации представляет сложности (окружность с центром в точке *A* на рис. 5).

Влияние заниженных результатов измерений на оценку точности локации

Время от времени трансиверы nanoLOC^{TM} измеряют расстояние с ошибкой (по нашим экспериментальным оценкам это менее 5 % измерений). В некоторых редких случаях зарегистрированное расстояние даже меньше действительного.

На рис. 6 показано, как использование окружности, соответствующей заниженному измерению, дает в результате расчетов неправильную уменьшенную область локации объекта.

При такого рода ошибках в измерениях расстояний от БС до объекта в большинстве случаев значение оценки точности *area-based*-локации (площадь области) будет меньше действительного (рис. 6), а сама область стопроцентного нахожде-

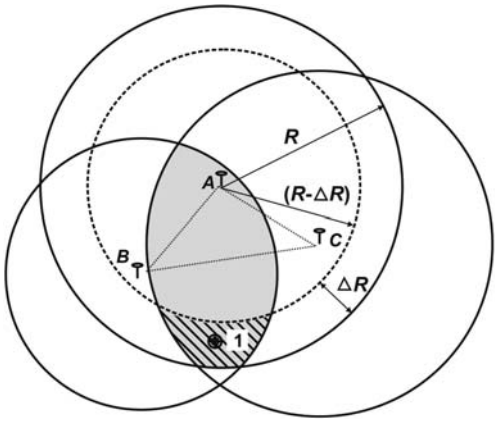


Рис. 4. Возможные конфигурации области локации

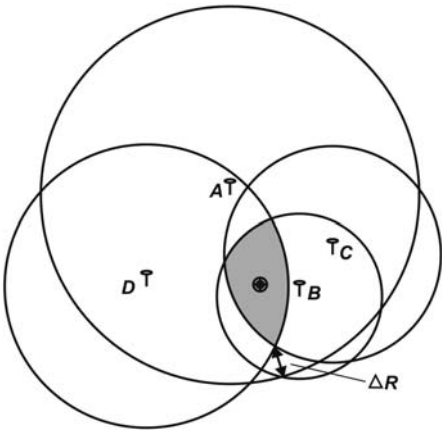


Рис. 5. Завышенный результат измерения от БСА

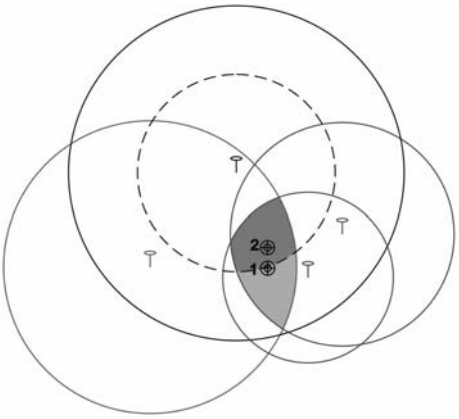


Рис. 6. Искажения результатов расчета локаций вследствие ошибок в измерении расстояний

ния объекта получается "смещенной". Это смещение области локации, несмотря на то, что значение оценки точности *area-based*-локации будет меньше (точность якобы выше), может в действительности увеличить расстояние между действительным положением объекта и рассчитанным на основе этой области — значение оценки точности *point-based*-локации будет больше, что соответствует меньшей точности.

При наличии результатов одного цикла измерений распознать такую ошибку чаще всего невозможно, кроме случая возникновения непересекающихся и не вложенных друг в друга окружностей соответствующих измерений. В таких случаях, очевидно, одна из окружностей соответствует заниженным измерениям.

Интегральная точность расчета локаций при многократном измерении расстояний до покоящегося объекта

Под интегральной точностью расчета локаций при наличии данных нескольких циклов измерений можно понимать значение оценки среднего значения точности локации по всем циклам. В серии циклов всегда можно рассчитать некое среднее (например "центр масс") и определить размер области или окружности, в которую с заранее заданной вероятностью попадает каждое рассчитанное значение локации (рис. 7).

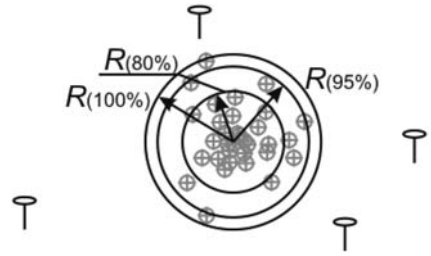


Рис. 7. Интегральная точность расчета локаций при многократном измерении расстояний до покоящегося объекта

Чем выше выбирается вероятность, тем больше будет радиус окружности и меньше уровень значимости оценки. Зависимость между значением вероятности и радиусом — нелинейная.

Для систем локаций, построенных на основе технологии nanoLOC^{TM}, как уже было сказано выше, можно выделить зону стопроцентного нахождения объекта. Для многих других технологий, где, например, измерение расстояний основано на измерении силы входного сигнала, выделить такую зону часто невозможно.

Интегральную оценку точности можно использовать вместо аналогичных *area-based*-оценок, поскольку она является более устойчивой к выбросам. Кроме того, она является простой с точки зрения вычислений, а способ ее расчета не зависит от используемых алгоритмов расчета локации по набору измеренных расстояний от БС, что позволяет применять ее для сравнения точности различных алгоритмов локаций.

Зависимость интегральной точности расчета локаций от числа измерений

Очевидно, что для получения более точной локации необходимо провести как можно больше замеров до как можно большего числа БС. В беспроводных сетях датчиков это не всегда возможно в силу ряда ограничений следующих параметров: ширины полосы частот, количества частотных диапазонов, числа замеров в секунду и т. п.

Предположим, что в зоне слышимости мобильного узла находятся пять БС. Пусть в целях экономии эфира в системе стоит ограничение на число измерений от БС для расчета одной локации, равное четырем. В зависимости от выбора четырех БС из пяти возможных, которые будут участвовать в измерениях, определение локаций может дать различные результаты (рис. 8).

На рис. 8, *а* показана ситуация, когда результатом расчета локации является точка *1* (есть измерение от базовой станции *A*, а от базовой станции *E* измерение отсутствует), а на рис. 8, *б* — около точки *2*. Таким образом, после накопления нескольких циклов измерений с различными набора-

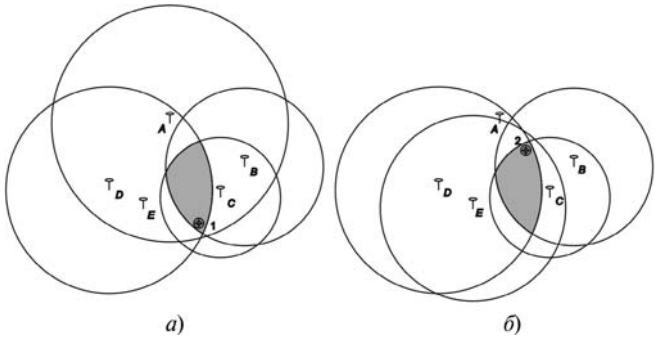


Рис. 8. Влияние на область локации выбора БС для измерения

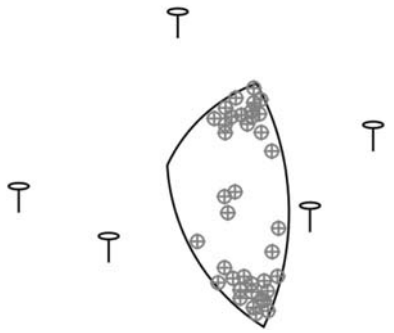


Рис. 9. Различие результатов определения локаций в зависимости от выбора базовых станций для измерения

ми БС в области вероятного нахождения объекта образуется два кластера точек (рис. 9).

Такие ситуации на практике встречаются очень часто. И хотя разброс и радиус зоны надежного определения местоположения в каждом из этих двух случаев невелики, это негативно влияет на восприятие человеком точности локации при наблюдении текущей локации в режиме *online* (точка, соответствующая текущему положению, "прыгает по карте").

Проверка точности расчета локаций

В современных технологиях сетей датчиков имеет смысл обсуждать методы расчета локации только для двумерной (в отличие от 3D) геометрии местности. В таких условиях можно проверить точность определения местоположения объекта, если знать с заранее заданной и достаточной точностью места расположения базовых станций и действительное местоположение статического узла, для которого проводится накопление результатов.

На практике [4, 9] для оценки точности локации часто используется эмпирическая функция распределения вероятности ошибки локации (рис. 10).

Для нахождения такой функции определяется разница ΔR_{err} между измеренным значением и действительным положением радиоузла до БС. Эта разница соответствует значению абсолютной погрешности одного измерения. После накопления достаточного количества данных строится функция, по оси абсцисс которой откладывается значение ΔR_{err}, равное разности между истинным и рассчитанным положениями объекта, а по оси ординат — значение, равное доле измерений, для которых абсолютное значение ошибки измерения ΔR_{err} меньше заданного значения R. В качестве точности локации используется величина, соответствующая ошибке ΔR_{err} при заданной вероятности. Часто в качестве точности локации используют 50- и 75 %-ные квантили соответствующих эмпирических функций распределения [9].

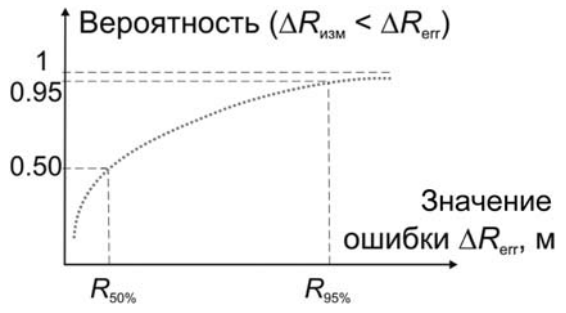


Рис. 10. Функция распределения вероятности ошибки локации

В ходе многочисленных экспериментов, выполненных за последние 2 года, авторы использовали все описанные выше подходы и методы для оценки точности локации в разрабатываемой ими системе определения локаций.

Как показала практика, для коммерчески обоснованной конфигурации базовых станций на открытом пространстве с плотностью расположения на местности до 1 шт/1000 м^2 возможно достижение абсолютной точности позиционирования объекта до 1—2 м. Для закрытых помещений в пределах зданий (плотность расположения БС в несколько раз выше) имеет смысл говорить лишь о надежном определении местоположения с точностью до комнаты.

Отметим, что для этих двух крайних случаев (открытые и закрытые пространства) для описания точности локации чаще всего применяют разные подходы — *point-based* и *area-based* соответственно.

Данное исследование проведено в рамках работы над проектами компании ООО "РТЛ-Сервис", а также при поддержке IT-парка Петрозаводского государственного университета и научно-образовательного центра по фундаментальным проблемам приложений физики низкотемпературной плазмы (проект RUX0-000013-PZ-06, Министерство образования и науки РФ, Правительство Республики Карелия, CRDF).

Список литературы

1. **Real Time** Location Systems (RTLS), Nanotron Technologies GmbH, Berlin, Germany, White paper NA-06-0248-0391-1.02. Apr. 2007.

2. **NanoLOC** Development Kit 3.0 © 2010 Nanotron Technologies GmbH | All rights reserved Subject to change without notice. April 2010 · NA-08-S-0016-E-2.3

3. **Мощевикин А. П., Галов А. С., Волков А. С.** Локация в беспроводных сетях датчиков стандарта nanoLOC (IEEE 802.15.4a) // Информационные технологии. 2011. № 8. С. 43—47.

4. **The Limits** of Localization Using Signal Strength: A Comparative Study / Eiman Elnahrawy, Xiaoyan Li, and Richard P. Martin // In IEEE SECON. P. 406—414, Santa.

5. **Гоголев А., Екимов Д., Екимов К., Мощевикин А., Федоров А., Цыкунов И.** Точность определения расстояний с помощью технологии nanoLOC // Беспроводные технологии. 2008. № 3 (12). С. 48—51.

6. **Дмитриев С., Екимов К., Кипрушкин С., Мощевикин А.** Изучение возможности применения технологии nanoLOC // Беспроводные технологии. 2008. № 3 (12). С. 52—56.

7. **Li Cong, Weihua Zhuang.** Non-Line-of-Sight Error Mitigation in Mobile Location // IEEE Transactions on Wireless Communications. March 2005. Is. 4, N 2. P. 560—573.

8. **Langley R. B.** Dilution of Precision // GPS World. 1999. May. P. 52—59.

9. **Chandrasekaran G., Ergin M., Yang J.** et al. Empirical Evaluation of the Limits on Localization Using Signal Strength Beyond Cramér-Rao Bounds // Proc. of IEEE SECON 2009. June 2009.

УДК 004.89 + 004.021

П. В. Казаков, канд. техн. наук, доц.,
Брянский государственный
технический университет,
e-mail: pvk_mail@list.ru

Оценка эффективности генетических алгоритмов многокритериальной оптимизации. Ч. 2*

На основе изложенных в части 1 статьи способов оценки качества работы генетических алгоритмов многокритериальной оптимизации (МГА) рассматривается практический пример анализа масштабируемости эффективности двух наиболее применяемых сейчас МГА при решении задач многокритериальной оптимизации разной сложности.

Ключевые слова: *многокритериальная оптимизация, принципы Парето, граница Парето, индикаторы эффективности, многокритериальные генетические алгоритмы SPEA2, NSGA-II*

Введение

В настоящее время генетические алгоритмы [1] являются одним из наиболее эффективных средств решения задач многокритериальной оптимизации (МО). Анализ статистики использования многокритериальных генетических алгоритмов (МГА) для решения задач МО позволяет сделать заключение об уровне их сложности, а также о наиболее часто используемых при этом МГА. Сейчас среди задач, решаемых с применением МГА, около 68 % являются двухкритериальными, 18 % — трехкритериальными и лишь 14 % имеют число критериев больше трех [2]; самыми используемыми при этом являются МГА второго поколения SPEA2 [3], NSGA-II [4]. Эти МГА отличаются относительно простыми реализацией и настройкой, а также различными принципами поиска решений. Кроме того, эти МГА имеют реализации в ряде коммерческих и свободных библиотек программирования для решения задач оптимизации [5—7], а также являются обязательными участниками сравнительных испытаний с другими генетическими алгоритмами многокритериальной оптимизации. В то же время, практика применения SPEA2, NSGA-II и получаемые ими высокие результаты, как правило, ограничиваются 2-, 3-критериальными задачами МО. Поэтому представляет интерес "независимая" оценка масштабируемости этих МГА при решении задач многокритериальной оптимизации большей размерности. В связи с этим было проведено комплексное исследование работы SPEA2, NSGA-II при решении набора из специальных тестовых задач. Были выбраны соответствующие индикаторы эффективности [см. часть 1], методика проведения испытаний, сделаны необходимые выводы.

1. Тестовые задачи для исследования МГА

Для оценки эффективности МГА используют специальные тестовые задачи. Они представляют собой обобщенные математические модели, в которых моделируются различные проблемы, которые могут возникнуть у МГА в реальных прикладных задачах МО. В частности, это отсутствие сведений о выпуклости/вогнутости пространства критериев, его дискретность и неравномерность, а также наличие ложных и изолированных экстремумов. Очевидно, что надежный МГА должен эффективно решать задачу при любых их сочетаниях.

К настоящему времени разработаны различные тестовые задачи для исследования эффективности МГА [8—12]. Они отличаются числом критериев (обычно 2—3), переменных оптимизации, а также характером проблемы, относительно которой проверяется МГА. Для унификации процедуры тестирования МГА одними из основоположников направления MOEA (*multi-objective evolutionary algorithms*) в эволюционных вычислениях была создана методика для автоматизации конструирования таких задач [8]. Ее главная идея заключается в том, что вначале аналитически определяется глобальная граница Парето, которая затем "помещается" в пространство критериев любой топологии. Это позволяет априорно знать лучшее решение и сравнивать с ним результаты МГА. На основе этой методики могут быть сформированы специальные тестовые наборы задач МО, позволяющие всесторонне оценить возможности испытуемого генетического алгоритма. Первая версия наиболее известного такого набора [9] состояла из шести двухкритериальных задач ZDT1—ZDT6 с буквенными префиксами от фамилий его авторов (Zitzler-Deb-Thile). Главным недостатком этого набора была немасштабируемость входящих в него задач — их исключительно двухкритериальность. Кроме того, генетические алгоритмы SPEA2, NSGA-II достаточно легко справлялись с этими задачами. Поэтому совместно

* Часть 1 статьи опубликована в № 8, 2012 г.

с Лауманнсом (*Laumanns*) была создана новая версия тестового набора DTLZ1—DTLZ9 [10]. Сейчас он считается стандартом де-факто для исследования МГА и содержит существенно усложненные задачи МО, характеризующиеся:

- масштабируемостью сложности благодаря возможности варьирования числа критериев и переменных оптимизации;
- известной информацией о глобально оптимальном множестве Парето;
- наличием методики модификации и создания новых тестовых задач различной сложности.

Анализ особенностей тестового набора DTLZ позволил без потери качества исследования эффективности МГА ограничить множество задач до DTLZ1, DTLZ2, DTLZ3 и DTLZ6. Каждая из них имеет характер минимизации и в совокупности они позволяют смоделировать все отмеченные проблемы на пути поиска решений. Во всех задачах число критериев $m \geqslant 2$, а число переменных оптимизации определяется как $n = k + m - 1$, где k — управляющий параметр сложности поискового пространства. Далее в качестве его значения используются значения, рекомендованные разработчиками задач.

DTLZ1

$$f_1(x) = \frac{1}{2} x_1 x_2 ... x_{m-1}(1 + g(x_h));$$

$$f_2(x) = \frac{1}{2} x_1 x_2 ... (1 - x_{m-1})(1 + g(x_h));$$

$$\vdots$$

$$f_{m-1}(x) = \frac{1}{2} x_1 (1 - x_2)(1 + g(x_h));$$

$$f_m(x) = \frac{1}{2} (1 - x_1)(1 + g(x_h)),$$

где $g(x_h) = 100(|x_h| + \sum_{x_i \in x_h} (x_i - 0{,}5)^2 - \cos(20\pi(x_i - 0{,}5)))$;

$$0 \leqslant x_i \leqslant 1,\ i = 1, ..., n,\ x_h \subset x;\ k = |x_h| = 5.$$

Глобальное множество Парето соответствует $x_h^* = \{0, 0, ..., 0\}$ и $\sum_{j=1}^{m} f_j = 0{,}5$.

DTLZ2

$$f_1(x) = (1 + g(x_h))\cos(x_1\pi/2)\cos(x_2\pi/2)...$$
$$...\cos(x_{m-2}\pi/2)\cos(x_{m-1}\pi/2);$$
$$f_2(x) = (1 + g(x_h))\cos(x_1\pi/2)\cos(x_2\pi/2)...$$
$$...\cos(x_{m-2}\pi/2)\sin(x_{m-1}\pi/2);$$
$$f_3(x) = (1 + g(x_h))\cos(x_1\pi/2)\cos(x_2\pi/2)...\sin(x_{m-2}\pi/2);$$
$$\vdots$$
$$f_{m-1}(x) = (1 + g(x_h))\cos(x_1\pi/2)\sin(x_2\pi/2);$$
$$f_m(x) = (1 + g(x_h))\sin(x_1\pi/2),$$

где $g(x_h) = \sum_{x_i \in x_h} (x_i - 0{,}5)^2$;

$$0 \leqslant x_i \leqslant 1,\ i = 1, ..., n,\ x_h \subset x;\ k = |x_h| = 10.$$

Глобальное множество Парето соответствует $x_h^* = \{0{,}5, 0{,}5, ..., 0{,}5\}$ и $\sum_{j=1}^{m} (f_j)^2 = 1$.

DTLZ3

$$f_1(x) = (1 + g(x_h))\cos(x_1\pi/2)\cos(x_2\pi/2)...$$
$$...\cos(x_{m-2}\pi/2)\cos(x_{m-1}\pi/2);$$
$$f_2(x) = (1 + g(x_h))\cos(x_1\pi/2)\cos(x_2\pi/2)...$$
$$...\cos(x_{m-2}\pi/2)\sin(x_{m-1}\pi/2);$$
$$f_3(x) = (1 + g(x_h))\cos(x_1\pi/2)\cos(x_2\pi/2)...\sin(x_{m-2}\pi/2);$$
$$\vdots$$
$$f_{m-1}(x) = (1 + g(x_h))\cos(x_1\pi/2)\sin(x_2\pi/2);$$
$$f_m(x) = (1 + g(x_h))\sin(x_1\pi/2),$$

где $g(x_h) = 100(|x_h| + \sum_{x_i \in x_h} (x_i - 0{,}5)^2 - \cos(20\pi(x_i - 0{,}5)))$;

$$0 \leqslant x_i \leqslant 1,\ i = 1, ..., n,\ k = |x_h| = 10.$$

Глобальное множество Парето соответствует $x_h^* = \{0{,}5, 0{,}5, ..., 0{,}5\}$ и $g^*(x_h) = 0$.

DTLZ6

$$f_1(x) = (1 + g(x_h))\cos(\theta_1\pi/2)\cos(\theta_2\pi/2)...$$
$$...\cos(\theta_{m-2}\pi/2)\cos(\theta_{m-1}\pi/2);$$
$$f_2(x) = (1 + g(x_h))\cos(\theta_1\pi/2)\cos(\theta_2\pi/2)...$$
$$...\cos(\theta_{m-2}\pi/2)\sin(\theta_{m-1}\pi/2);$$
$$f_3(x) = (1 + g(x_h))\cos(\theta_1\pi/2)\cos(\theta_2\pi/2)...\sin(\theta_{m-2}\pi/2);$$
$$\vdots$$
$$f_{m-1}(x) = (1 + g(x_h))\cos(\theta_1\pi/2)\sin(\theta_2\pi/2);$$
$$f_m(x) = (1 + g(x_h))\sin(\theta_1\pi/2),$$

где $g(x_h) = \sum_{x_i \in x_h} (x_i)^{0,1}$;

$$\theta_i = \frac{\pi}{4(1 + g(x_h))}(1 + 2g(x_h)x_i),\ i = 2, 3, ..., (m - 1);$$

$$0 \leqslant x_i \leqslant 1,\ i = 1, ..., n;\ x_h \subset x;\ k = |x_h| = 10.$$

Глобальное множество Парето соответствует $x_h^* = \{0, 0, ..., 0\}$; $g^*(x_h) = 0$; $\theta_i = \pi/4$, $i = 2, 3, ..., (m - 1)$.

2. Методика проведения исследований эффективности МГА

Основная цель проведения испытания различных МГА заключается в оценке степени их масштабируемости при росте сложности решаемых задач. В данном случае предполагается проверка возможности SPEA2, NSGA-II сохранять необходимую точность и приемлемую скорость вычислений при решении задач DTLZ1, DTLZ2, DTLZ3, DTLZ6 с разным числом критериев $m = \{2, 4, 6, 8\}$. Таким образом, каждому из тестируемых МГА предстоит решить 16 задач многокритериальной оптимизации. Учитывая, что изначально известно точное решение каждой задачи, эффективность МГА будет оцениваться по следующему набору индикаторов $\{I_{ONVG}$,

Таблица 1

Значения параметров работы МГА, зависящих от числа критериев

Число критериев	Размер популяции	Размер Парето-архива (размер популяции)	Число поколений	Число запусков МГА
2	100	0,28 (28)	300	30
4	250	0,45 (113)	500	30
6	400	0,52 (208)	700	10
8	600	0,6 (360)	1000	10

I_S, I_{DE}, I_{GD}, I_{OT}} [часть 1, см. журнал № 8, 2012 г.]. В общей сложности для последующего анализа с каждым МГА будет связано 16 × 5 = 80 количественных показателей.

Для каждого МГА должен быть определен набор значений управляющих параметров. Для одних это выполнялось в соответствии с рекомендациями, для других эмпирически. Известно, что в работе МГА ключевую роль играет размер популяции. В работе [13] были определены соотношения между числом критериев, размером популяции и максимальным числом недоминируемых решений в ней. Следуя полученным рекомендациям, были определены значения размера популяции и размера Парето-архива (доля от размера популяции) для разного числа критериев (табл. 1).

Время работы МГА во всех случаях ограничивалось только числом поколений. Относительно выбора этого значения не существует определенных рекомендаций для МГА. Однако экспериментально показано, что простое увеличение времени работы МГА не гарантирует роста недоминируемых решений, а в случае мультимодальности отдельных критериев приводит к снижению мощности итогового множества Парето [13]. Поэтому в данном случае число поколений выбирали с точки зрения сохранения разумной пропорции с размером популяции. Ради объективности испытаний МГА запускали по 30 раз для $m = \{2, 4\}$ и, учитывая резкий рост времени поиска, по 10 раз для $m = \{6, 8\}$. Для всех МГА и решаемых задач использовали турнирный отбор, одноточечный кроссинговер и одноточечную мутацию. Значения вероятностей операторов кроссинговера и мутации (p_c, p_m) выбирали из заданных интервалов $p_c \in [0{,}7, 0{,}9]$, $p_m \in [0{,}001, 0{,}01]$/бит соответственно. Для этого при решении каждой из задач, но только для $m = 2$ определяли значения p_c, p_m, при которых SPEA2, NSGA-II достигали лучшего результата по индикатору I_{GD}. Найденные таким образом значения вероятностей для каждого МГА использовали во всех остальных случаях. Это позволило упростить процедуру исследования МГА без потери объективности полученных результатов.

3. Полученные результаты и их анализ

Для удобства анализа результаты решения всех задач сгруппированы отдельно по каждому индикатору (табл. 2—6). Это позволит проследить динамику изменения соответствующих показателей алгоритмов при изменении задачи и ее сложности. Значения всех индикаторов усреднены по выполненному числу запусков МГА.

Анализ табл. 2 позволяет сделать вывод, что во всех случаях число найденных недоминируемых решений оказалось выше у NSGA-II. Также, в частности, можно отметить следующее.

- Для $m = 2$ у NSGA-II значение индикатора заметно больше, чем у SPEA2 во всех задачах. Это можно объяснить ограничением значения индикатора у SPEA2 размером Парето-архива, в то время как у NSGA размером популяции.
- Во всех задачах для $m = \{6, 8\}$ у SPEA2 значения индикатора меньше размера Парето-архива. Это может означать системное снижение разнообразия популяции на некотором этапе поиска.

Таблица 2

Результаты по индикатору I_{ONVG}

Задача	Число критериев	SPEA2	NSGA-II
DTLZ1	2	28	66
	4	105	168
	6	197	262
	8	254	389
DTLZ2	2	28	63
	4	112	179
	6	189	271
	8	247	385
DTLZ3	2	19	47
	4	93	127
	6	167	236
	8	237	364
DTLZ6	2	27	71
	4	108	173
	6	194	263
	8	269	401

Таблица 3

Результаты по индикатору I_{GD}

Задача	Число критериев	SPEA2	NSGA-II
DTLZ1	2	0,071	0,067
	4	0,312	0,574
	6	5,693	8,647
	8	387,241	324,724
DTLZ2	2	0,012	0,015
	4	5,732	6,124
	6	10,372	9,785
	8	17,935	12,352
DTLZ3	2	0,062	0,178
	4	5,371	27,657
	6	226,953	310,603
	8	1847,632	1641,533
DTLZ6	2	0,107	0,112
	4	5,093	5,727
	6	10,491	10,475
	8	16,623	11,237

Таблица 4

Результаты по индикатору I_S

Задача	Число критериев	SPEA2	NSGA-II
DTLZ1	2	0,186	0,223
	4	0,201	0,206
	6	0,292	0,368
	8	0,401	0,413
DTLZ2	2	0,112	0,119
	4	0,133	0,124
	6	0,373	0,286
	8	0,434	0,371
DTLZ3	2	0,121	0,147
	4	0,134	0,245
	6	0,328	0,287
	8	0,459	0,488
DTLZ6	2	0,097	0,105
	4	0,172	0,154
	6	0,185	0,171
	8	0,251	0,201

Таблица 5

Результаты по индикатору I_{DE}

Задача	Число критериев	SPEA2	NSGA-II
DTLZ1	2	0,971	0,964
	4	1,283	1,381
	6	1,141	1,658
	8	1,412	1,486
DTLZ2	2	1,371	1,380
	4	1,714	1,847
	6	2,141	2,317
	8	2,113	2,245
DTLZ3	2	1,386	1,375
	4	1,712	1,894
	6	2,137	2,312
	8	2,114	2,206
DTLZ6	2	1,171	1,167
	4	1,541	1,673
	6	1,816	1,977
	8	1,862	1,913

Таблица 6

Результаты по индикатору I_{OT} (секунды)

Задача	Число критериев	SPEA2	NSGA-II
DTLZ1	2	6,3	4,8
	4	174,2	28,1
	6	3357,4	353,7
	8	67114,5	2577,2
DTLZ2	2	6,8	5,4
	4	191,4	31,6
	6	3689,6	397,6
	8	73771,8	2904,4
DTLZ3	2	11,2	9,4
	4	312,1	56,1
	6	6217,3	698,7
	8	119324,2	5094,3
DTLZ6	2	7,2	5,3
	4	197,8	30,6
	6	3843,7	390,2
	8	76784,3	2843,8

- В задаче DTLZ3 оба МГА нашли наименьшее число недоминируемых решений. Это может быть связано с попаданием в одну из множества локальных границ Парето.

Анализ результатов по индикатору I_{GD} (табл. 3) показывает, что в задачах DTLZ1, DTLZ2, DTLZ6 для $m = \{2, 4\}$ SPEA2, NSGA-II получили близкие значения. Остальные случаи можно охарактеризовать следующим образом.

- При $m = \{6, 8\}$ в задачах DTLZ1 и DTLZ3 оба МГА продемонстрировали достаточно плохой результат, особенно при $m = 8$. Пространства критериев этих задач имеют множество $(11^k - 1)$ для DTLZ1 и $(3^k - 1)$для DTLZ3 локальных границ Парето. Очевидно, при росте числа критериев у SPEA2, NSGA-II оказывается недостаточно возможностей для исследования всего поискового пространства.
- У NSGA-II при $m = 8$ значения индикатора во всех случаях оказались лучше, чем у SPEA2.

В целом по индикатору I_S (табл. 4) у обоих МГА оказались относительно близкие значения. В задачах DTLZ1, DTLZ3 в большинстве случаев лучшие результаты у SPEA2, а DTLZ2 и DTLZ6 у NSGA-II.

Анализ полученных значений индикатора I_{DE} (табл. 5) показывает, что наиболее полный охват границы Парето по всем размерностям был получен обоими МГА только для $m = 2$. С ростом числа критериев качество поиска по этому индикатору у исследуемых МГА снижается по-разному. В частности:

— для задачи DTLZ1 обоими МГА был достигнут лучший результат;

— в большинстве случаев NSGA-II превзошел SPEA2, прежде всего, благодаря большему числу найденных Парето-оптимальных решений;

— для $m = 8$ результат обоих МГА достаточно низкий.

Время работы МГА определялось в одинаковых условиях на одной конфигурации компьютера. Из табл. 6 видно, что с ростом числа критериев время вычислений резко увеличивается. Важным оказалось, что во всех случаях NSGA-II оказался существенно быстрее SPEA2.

Обобщив значения, полученные SPEA2, NSGA-II по каждому из индикаторов, можно оценить способность этих генетических алгоритмов сохранять эффективность при усложнении решаемых задач МО. Далее приведены соответствующие выводы, объединенные по трем ранее названным основным критериям эффективности МГА [см. часть 1].

1. Сходимость к оптимальным решениям. SPEA2, NSGA-II сумели достаточно точно решить тестовые задачи, но не во всех случаях. Сходимость МГА имела тенденцию к ухудшению при $m > 4$. Причем эта тенденция сохранялась независимо от изначальной сложности самого поискового пространства. Причина этого видится в том, что с ростом числа критериев увеличивалась скорость заполнения популяции недоминируемыми решениями. В итоге уже в начале работы МГА все хромосомы популяции имели одинаковую пригодность, что в итоге приводило к резкой стагнации поиска. Как и в однокритериальном случае существенно усложняет поиск наличие множества локальных оптимумов (границ Парето). При решении такой задачи DTLZ3 все полученные результаты оказались хуже остальных.

2. Протяженность и равномерность заполнения границы Парето. В целом, полученные значения соответствующих индикаторов у SPEA2, NSGA-II подтверждают эффективность заложенных в них принципов сохранения разнообразия найденных Парето-оптимальных решений. В то же время для $m > 6$ результаты у обоих МГА ухудшаются — число обрабатываемых хромосом становится недостаточным для аппроксимации границы Парето. NSGA-II во всех тестовых испытаниях показал по индикатору I_{DE} результаты лучше, чем у SPEA2. В связи с этим можно сделать предположение о неэффективности сохранения в процессе поиска неизменных пропорций между размером популяции и Парето-архивом.

3. Время поиска. Во всех случаях NSGA-II превзошел SPEA2, причем разница становится критической с ростом m. Дополнительные операции, связанные с обработкой Парето-архива, привели к экспоненциальному увеличению времени поиска у SPEA2.

Заключение

Таким образом, SPEA2, NSGA-II можно назвать относительно масштабируемыми, в частности для $m < 6$. С ростом числа оптимизируемых критериев

вероятность нахождения данными МГА глобальной границы Парето существенно снижается, так же как плотность ее аппроксимации и протяженность. Что касается времени, затраченного на поиск, то при увеличении $m > 6$ использование NSGA-II становится более предпочтительным. Вместе с тем, полученные результаты не позволяют однозначно назвать лучший из этих двух МГА. У каждого из них есть задачи и индикаторы, по которым SPEA2, NSGA-II превзошли друг друга. В целом это подтверждает практику их самостоятельного использования при решении задач МО с $m < 4$, в остальных случаях наиболее оправданным видится их совместное применение.

Важно подчеркнуть, что полученные результаты следует воспринимать лишь как общий потенциал масштабируемости главных современных генетических алгоритмов многокритериальной оптимизации. Также проведенные исследования направлены не на выявление достоинств и недостатков SPEA2, NSGA-II, а скорее на определение перспектив их совершенствования. В настоящее время к таковым можно отнести использование различных вариантов генетических операторов, интеграции с альтернативными метаэвристиками, распределенной среды вычислений.

Список литературы

1. **Гладков Л. А., Курейчик В. В., Курейчик В. М.** Генетические алгоритмы. М.: ФИЗМАТЛИТ, 2006. 320 с.
2. **List of References** on Evolutionary Multiobjective Optimization. URL: http://delta.cs.cinvestav.mx/~ccoello/EMOO/EMOObib.html, http://delta.cs.cinvestav.mx/~ccoello/EMOO/EMOOstatistics.html.
3. **Zitzler E., Laumanns M., Thiele L.** SPEA2: Improving the Strength Pareto Evolutionary Algorithm. // EUROGEN 2001. Evolutionary Methods for Design, Optimization and Control with Applications to Industrial Problems. 2002. P. 95—100.
4. **Deb K., Pratap A., Agarwal S., Meyarivan T.** A Fast and Elitist Multiobjective Genetic Algorithm: NSGA—II // IEEE Transactions on Evolutionary Computation. 2002. N 6 (2). P. 182—197.
5. **KEA** (Kit for Evolutionary Algorithms). URL: http://ls11-www.cs.uni-dortmund.de/people/schmitt/Daten/Kea/kea.jsp.
6. **PISA** (Platform and Programming Language Independent Interface for Search Algorithms). URL: http://www.tik.ee.ethz.ch/pisa/.
7. **Optimizing** Products Performance with Finite Element Analysis & Multiphysics Simulation. URL: http://www.simulia.com/products/isight.html.
8. **Deb K.** Multi-objective genetic algorithms: problem difficulties and construction of test problems // Evolutionary computation. 1999. N 7 (3). P. 205—230.
9. **Zitzler E., Deb K., Thiele L.** Comparison of Multiobjective Evolutionary Algorithms: Empirical Results // Evolutionary Computation. 2000. N 8 (2). P. 173—195.
10. **Deb K., Thiele L., Laumanns M., Zitzler E.** Scalable Test Problems for Evolutionary Multi-Objective Optimization. Evolutionary Multiobjective Optimization. Theoretical Advances and Applications. Springer. 2005. P. 105—145.
11. **Okabe T., Jin Y., Olhofer M., Sendhoff B.** On Test Functions for Evolutionary Multiobjective Optimization // Proc. of Parallel Problem Solving form Nature — PPSN VIII. 2004. P. 792—802.
12. **Huband S., Hingston P., Barone L., While L.** A Review of Multiobjectives Test Problems and a Scalable Test Problem Toolkit // IEEE Transactions on Evolutionary Computation. 2006. N 10 (5). P. 477—506.
13. **Deb K.** Multi-Objective Optimization Using Evolutionary Algorithms. Wiley, 2009. 536 p.

УДК 519.612

К. Ф. Иванова, канд. техн. наук,
e-mail: klara.i2010@yandex.ru,
Санкт-Петербургский государственный университет

Знаковый подход к оценке решения интервальных линейных систем

Предлагается новый алгебраический подход к оценке решения интервальной линейной системы, реализованный на основе "знаковой" методики, при котором исходная задача заменяется точечными (неинтервальными) системами в евклидовом пространстве. Конструируется специализированный алгоритм, позволяющий выполнить покомпонентную оценку вектора неизвестных для точечных систем, аналогичную по результатам "внешней" оценки множества решений, получаемых методами интервальной алгебры. Применение "знаковой" методики совмещает высокую вычислительную эффективность с высоким качеством оценивания множества решений при выполнении одновременно оценки чувствительности линейных интервальных систем.

***Ключевые слова**: "внешняя" оценка, "знаковая" методика, интервальная линейная система, множество решений, точечные системы*

Введение

Математическое моделирование, объектом исследования которого являются процессы, описываемые интервальными системами линейных алгебраических уравнений (ИСЛАУ), все более расширяет область своего приложения, включая изучение физических, экономических, экологических и социальных процессов. Решение задач, основанных на решении систем линейных уравнений, имеющих интервальную или ограниченную неопределенность, составляют большой пласт востребованной информации о поведении объекта, данные о котором получены в результате измерений. В силу этого в настоящее время существует большое число моделей и методов решения ИСЛАУ, позволяющих получить внешние и внутренние оценки множества решений, определяющих состояние исследуемого объекта. Основополагающие результаты в области интервального анализа и его приложений были получены в работах наших и зарубежных ученых: Л. В. Канторовича, А. Б. Куржанского, Ю. И. Шокина, С. П. Шарого, А. В. Лакеева, А. П. Вощинина, Н. М. Оскорбина, Г. Г. Меньшикова, Р. Мура,

Е. Хансена, Г. Алефельда, А. Неймайера, Ю. Рона и многих других исследователей.

Целью данной работы является постановка и решение задачи поиска внешнего интервала для множества решений ИСЛАУ и установление принадлежности решения некоторому компактному допустимому множеству, ограничивающему область неопределенности, инициированную неточными измерениями. Эта задача, по существу, — интервальная форма хорошо известной задачи о чувствительности для линейной системы, когда и вариации параметров и оценки вариаций решения рассматриваются в интервальном виде.

Предлагается новый алгебраический подход определения множества решений на основе "знаковой" методики определения чувствительности ИСЛАУ. Существуют многочисленные публикации по наиболее важным результатам вычисления внешних покоординатных оценок для множества решений ИСЛАУ, таких как интервальный метод Гаусса, процедура Хансена—Рона, методика Ньютона. Однако эта задача по-прежнему остается достаточно трудной и трудоемкой.

Идея метода "знаковой" методики в приложении к СЛАУ принадлежит Ю. П. Петрову [1], дальнейшее развитие которой нашло отражение в работе [7] и в данном исследовании.

В статье используются некоторые понятия и терминология интервальной математики, принятые в работах С. П. Шарого. Под объединенным множеством решений ИСЛАУ понимается множество, образованное всевозможными решениями точечных систем, матрицы коэффициентов и векторы правых частей которых принадлежат их интервальным аналогам. Под "внешней" оценкой множества решений ИСЛАУ понимаются "внешние" покоординатные оценки множества решений, образованных всеми решениями точечных систем или интервальный вектор-брус. Алгебраический подход решения интервальной системы, а именно представление ее системой с точечными матрицами, практикуется многими исследователями. Так, метод решения интервальной "внешней" задачи, впервые предложенный С. П. Шарым, состоит в замене исходной постановки интервальной системы на задачу решения одной точечной (неинтервальной) системы уравнений в евклидовом пространстве двойной размерности. При этом конструируется специализированный алгоритм — субдифференциальный метод Ньютона, реализующий новый подход. Анализировать субдифференциальный метод Ньютона и его дальнейшие усовершенствования другими авторами не является областью нашей компетенции и нашей задачей. Можно только отметить, что использование этих подходов имеет определенные, иногда значительные ограничения на вид исходной интервальной матрицы. Из литературы известно также, что решение интервальной системы, особенно большой размерности, представляет собой архитрудную задачу. Поэтому оценка чувствительности СЛАУ от погрешности коэффициентов матрицы и правых частей по "знаковой" методике, оказалась весьма соблазнительной, чтобы использовать ее и для оценки решения интервальных линейных систем, если вариация коэффициентов исходной СЛАУ не приводит к вырожденности возмущенной матрицы.

Хотелось бы считать, что предлагаемый алгебраический подход оценки чувствительности интервальных линейных систем является одной из новых реализаций алгебраического подхода к решению ИСЛАУ, при котором исходная интервальная система сводится к решению точечных уравнений в евклидовом пространстве.

В тексте принята система обозначений, следующая, главным образом, тем неофициальным международным рекомендациям, которые выработаны специалистами по интервальному анализу. Интервалы и интервальные объекты (векторы, матрицы) обозначаются жирным шрифтом (например, **A**, **b**), тогда как неинтервальные (точечные) величины никак специально не выделяются. Вместо принятого подчеркивания и надчеркивания объектов, означающих взятие нижнего и верхнего концов интервала, для удобства изложения нами введены символы (+) и (−) соответственно.

Символьные обозначения:

R — множество всех вещественных чисел;

IR — множество всех интервалов;

$\mathbf{IR}^n$ — множество n-мерных векторов с элементами из **IR**;

$\mathbf{IR}^{n \times n}$ — множество $n \times n$ матриц с элементами из **IR**;

A, **b**, **x** — интервалы и интервальные объекты (векторы, матрицы);

$\Sigma_{EE}(\mathbf{A}, \mathbf{b})$ — объединенное множество решений, образованное решением точечных систем:

A, b, $\overline{\mathrm{A}}$, $\overline{\mathrm{b}}$, x — точечные матрицы и векторы;

A_{ij} — алгебраическое дополнение к элементу a_{ij};

(+) и (−) — индексация точечных граничных объектов, полученных в сторону наибольшего уменьшения и увеличения;

A^- и A^+ — граничные точечные матрицы и матрицы, получившие максимальное линейное изменение определителя матрицы A в сторону его уменьшения и увеличения;

a_{ij}^- и a_{ij}^+ — элементы точечных матриц A^- и A^+;

b_{ij}^- и b_{ij}^+ — компоненты вектора правых частей точечных матриц, соответствующие приращению детерминанта при максимальном уменьшении и увеличении;

a_-^j, b_-^j и a_+^j, b_+^j — j-е столбцы коэффициентов точечных матриц компонент A^- и A^+ правых частей линейных систем, полученные в сторону уменьшения и увеличения определителей матриц;

x^- и x^+ — векторы решения точечных систем уравнений с матрицами и правыми частями, обеспечившими наибольшее и наименьшее их изменение;

Δ — определитель исходной матрицы точечных измерений для СЛАУ;

$\overline{A} = \mathrm{mid}(\mathbf{A})$, $\overline{b} = \mathrm{mid}(\mathbf{b})$ — средние значения матрицы и вектора правой части интервальной системы;

Δ_- и Δ_+ — определители точечных матриц, полученные с приращениями в сторону их убывания и возрастания соответственно;

Δ^{b^j} — определитель с замещением вектором правой части j-го столбца в исходной матрице.

Примеры возможных подстановок вектор-столбцов в матрицы при вычислении j-й компоненты вектора x:

$\Delta_-^{b_+^j}$ — подстановка столбца свободных членов b_+^j, полученного в сторону возрастания определителя расширенной матрицы A^+, в матрицу A^- при вычислении детерминанта матрицы A в сторону его уменьшения;

$\Delta_+^{a_-^j}$ — постановка столбца a_-^j из матрицы A^-, полученной в сторону уменьшения детерминанта A в матрицу A^+, но полученную в сторону увеличения ее детерминанта.

* * *

Напомним, что интервальная система является системой линейных уравнений

$$\mathbf{Ax} = \mathbf{b} \qquad (1)$$

с интервальной $n \times n$-матрицей $\mathbf{A} = (\mathbf{a}_{ij}) \in \mathbf{IR}^{n \times n}$ и интервальным n-вектором правой части. $\mathbf{b} = (b_i) \in \mathbf{IR}^n$. Интервальная система (1) мыслится как совокупность всех точечных $n \times n$-систем

$$\mathrm{Ax} = \mathrm{b} \qquad (2)$$

с точечной матрицей $\mathrm{A} = (a_{ij}) \in \mathbf{R}^{n \times n}$ и правой частью $\mathrm{b} = \{b_i\}$, $i, j = 1, 2, ..., n$; x — искомый вектор с координатами x_j, $\mathrm{x} = \{x_j\}$, $\mathrm{x}, \mathrm{b} \in \mathbf{R}^n$. Решение такой системы существует и единственно, если матрица A невырожденная.

Постановка интервальной задачи заключается в определении покоординатной оценки множества решений

$$\Sigma_{\mathrm{EE}}(\mathbf{A}, \mathbf{b}) = \{\mathrm{x} \in \mathbf{R}^n | (\exists \mathrm{A} \in \mathbf{A})(\exists \mathrm{b} \in \mathbf{b})(\mathrm{Ax} = \mathrm{b})\}, \quad (3)$$

образованного всеми решениями точечных систем $\mathrm{Ax} = \mathrm{b}$ с $\mathrm{A} \in \mathbf{A}$ и $\mathrm{b} \in \mathbf{b}$ или оценке $\min\{x_k | \mathrm{x} \in \Sigma_{\mathrm{EE}}\}$ снизу и $\max\{x_k | \mathrm{x} \in \Sigma_{\mathrm{EE}}\}$ сверху для $k = 1, 2, ..., n$.

Формулировка интервальной "внешней" задачи предполагает поиски всего возможного внешнего интервала для объединенного множества решений (3) системы (1) и является одной из классических форм ее постановки [2, 3].

Считается при этом, что интервальная матрица неособенная и все составляющие ее точечные матрицы также неособенные. Если интервалы являются вырожденными, то все вычислительные операции становятся операциями над вещественными числами.

Предлагаемый алгебраический подход, направленный на определение покоординатного решения ИСЛАУ, состоит в замещении интервальной системы "внешней" задачи двумя вещественными системами на основании "знаковой" методики. Согласно [2] отображение $\mathbf{IR}^n \to \mathbf{R}^n$ или вложение интервального пространства $\mathbf{IR}^n$ в линейное пространство $\mathbf{R}^n$ является взаимно однозначным, что правомерно и для предлагаемого подхода. Реализацию основной идеи перехода от интервальной к точечной линейной системе можно представить следующими ступенями:

1. Проверяется неособенность квадратной матрицы $n \times n$ изучаемой ИСЛАУ.

2. Компоненты матрицы интервальной системы представляются двумя границами — нижней и верхней, симметричными относительно своего среднего значения. По существу, такие компоненты можно понимать как некие реальные измерения $\overline{a}_{ij}$, имеющие относительные погрешности отклонения ε_{ij} влево и вправо от среднего $\overline{a}_{ij} - \varepsilon_{ij}\overline{a}_{ij} < \overline{a}_{ij} < \overline{a}_{ij} + \varepsilon_{ij}\overline{a}_{ij}$.

3. Выполняется переход от интервальной матрицы к точечной выбором одной из границ интервала (левой или правой) для каждого элемента матрицы, проводимым по "знаковой" методике, тем самым создавая условия априорного получения одной из крайних (угловых) компонент вектора решения.

4. Выбор экстремальных значений среди всех компонент векторов, полученных по предлагаемой методике, обеспечивает нам ожидаемый разброс (объединенного) множества решений, ограниченного гиперпараллелипипедом крайних значений компонент.

Основная идея предлагаемого алгебраического подхода заключается в определении знаков задаваемых относительных погрешностей коэффициентов исходной матрицы и компонент вектора правых частей, при которых возникает максимальная погрешность вектора решения СЛАУ. Для линейных алгебраических систем исходная матрица — это матрица, полученная в результате измерений, которая приводит к неточному результату решения. Для интервальных систем в качестве исходной матрицы и правой части выбираются средняя матрица $\overline{A} = \mathrm{mid}(\mathbf{A})$ и средний вектор $\overline{b} = \mathrm{mid}(\mathbf{b})$. Тогда относительные погрешности, определяющие отклонения влево и вправо от средних значений, получаются элементарными алгебраическими преобразованиями:

$$\varepsilon_{ij} = \frac{(\overline{a}_{ij} - a_{ij}^{left})}{\overline{a}_{ij}} = \frac{(a_{ij}^{right} - \overline{a}_{ij})}{\overline{a}_{ij}};$$

$$\delta_i = \frac{(\overline{b}_i - b_i^{left})}{\overline{b}_i} = \frac{(b_i^{right} - \overline{b}_i)}{\overline{b}_i}, \qquad (4)$$

что позволяет выразить интервальные коэффициенты в виде

$$\boldsymbol{a}_{ij} \in [\overline{a}_{ij} - \varepsilon_{ij}\overline{a}_{ij},\ \overline{a}_{ij} + \varepsilon_{ij}\overline{a}_{ij}];$$
$$\boldsymbol{b}_i \in [\overline{b}_i - \delta_i\overline{b}_i,\ \overline{b}_i + \delta_i\overline{b}_i];\ \forall i = \overline{1, n},\ \forall j = \overline{1, n}, \quad (5)$$

которые можно рассматривать как интервалы для неточных измерений коэффициентов матрицы с учетом попадания их точных значений в возможные интервалы измерения.

Создание алгоритма на основе "знаковой" методики базируется на определении знаков относительных погрешностей элементов и компонент правых частей (левой или правой границы), приводящих к максимальным погрешностям компонент вектора решения [1,6]. Этот подход следует из анализа сочетания знаков элементов матрицы $a_{ij} + \varepsilon_{ij}a_{ij}$ и соответствующих им алгебраических дополнений A_{ij}, обеспечивающих вычисление определителя системы. Главную линейную часть приращения определителя можно представить как сумму произведений элементов столбца матрицы на их алгебраические дополнения и на относительные погрешности ε_{ij}:

$$\Delta_{\text{лин}} = \sum_{i=1}^{n} a_{ij}A_{ij}\varepsilon_{ij},\ \forall j = \overline{1, n}. \quad (6)$$

Отсюда следует, что наибольшее возможное значение (6) в линейном приближении для приращения определителя возможно в том случае, когда относительная погрешность ε_{ij} по знаку совпадает со знаком произведения $a_{ij}A_{ij}$.

Компоненты вектора правой части b_i с приращениями представляются аналогично компонентам матрицы как $b_i + \delta_i b_i$. Тогда при подстановке их в матрицу алгебраической системы главное линейное приращение определителя выразится суммой

$$\Delta_{\text{лин}} = \sum_{i=1}^{n} b_iA_{ij}\delta_i,\ \forall j = \overline{1\ldots n}. \quad (7)$$

Как и в первом случае, максимальное линейное приращение (7) получается тогда, когда знак δ_i совпадает со знаком произведения $b_i \cdot A_{ij}$. Если сомножители этого произведения оказываются одного знака, наибольшее приращение обеспечивается положительным значением δ_i, если разного — отрицательным. Полученные знаковые неравенства для погрешностей компонент матрицы и правых частей можно записать в общем виде:

$$\begin{aligned}
&A_{ij} > 0,\ a_{ij} > 0 \to \varepsilon_{ij} > 0;\ A_{ij} > 0,\ b_i > 0 \to \delta_i > 0;\\
&A_{ij} < 0,\ a_{ij} < 0 \to \varepsilon_{ij} > 0;\ A_{ij} < 0,\ b_i < 0 \to \delta_i > 0;\\
&A_{ij} > 0,\ a_{ij} < 0 \to \varepsilon_{ij} < 0;\ A_{ij} > 0,\ b_i < 0 \to \delta_i < 0;\\
&A_{ij} < 0,\ a_{ij} > 0 \to \varepsilon_{ij} < 0;\ A_{ij} < 0,\ b_i > 0 \to \delta_i < 0.
\end{aligned} \quad (8)$$

Формальный переход от интервальных коэффициентов (5) к их вещественным значениям (8) заключается в выборе только одной границы интервалов, расчетная реализация которых приводит к выявлению максимальной вариации "среднего" решения:

$$\begin{aligned}
&a_{ij}^{\pm} = [a_{ij} \pm a_{ij}\varepsilon_{ij}\operatorname{sgn}(a_{ij}A_{ij})],\\
&b_i^{\pm} = [b_i \pm b_i\delta_i\operatorname{sgn}(b_iA_{ij})],\\
&(+), \text{ если } \varepsilon_{ij},\ \delta_i \text{ и } \operatorname{sgn}(a_{ij}A_{ij}) \text{ одного знака;}\\
&(-), \text{ если } \varepsilon_{ij},\ \delta_i \text{ и } \operatorname{sgn}(a_{ij}A_{ij}) \text{ разного знака;}\\
&i = \overline{1, n},\ j = \overline{1, n}.
\end{aligned} \quad (9)$$

Эта операция осуществляется заменой измеренных интервальных элементов a_{ij} одним из граничных значений a_{ij}^- или a_{ij}^+, вызывающих наибольшие отклонения детерминанта (6), (7) в положительном или отрицательном направлении выбором знаков относительных погрешностей ε_{ij} из неравенств (8) [6]. Верхний символ (+) или (−) компонент (9) соответствует максимальному увеличению или уменьшению определителя и, в дальнейшем, получению решения в сторону верхних и нижних интервальных оценок. Точечные матрицы, сконструированные по такому правилу, определяют окружение для интервальной системы (1).

Из сказанного выше следует, что для применения "знаковой" методики к интервальной задаче "внешнего" оценивания необходимо, чтобы относительная погрешность элементов системы не превосходила 100 %. Только в этом случае линейное приращение матриц системы и построенный на этом дальнейший анализ безукоризненно работает. Для матриц с погрешностями коэффициентов, больших или равных 100 %, нельзя воспользоваться предложенной методикой оценки решения. Для нулевых средних значений следует рассматривать обе границы компоненты с дальнейшей оценкой детерминанта матрицы.

Исходя из точечного представления компонент интервальную матрицу и правую часть системы (1) можно представить двумя составляющими, заменяя их двумя вещественными матрицами с компонентами

$$\mathbf{A} = [A^-, A^+] \text{ и } \mathbf{b} = [b^-, b^+], \quad (10)$$

так что создаются две линейные алгебраические системы

$$\begin{cases} A^-x^- = b^-; \\ A^+x^+ = b^+, \end{cases} \quad (11)$$

где A^- и A^+ являются граничными точечными матрицами и векторами правых частей, принадлежащими интервальной системе: $A^- \in \mathbf{A}$ и $A^+ \in \mathbf{A}$, $b^- \in \mathbf{b}$ и $b^+ \in \mathbf{b}$ и такими, что $A^- = -A^+$, $-A^- = A^+$, $b^- = -b^+$, $-b^- = b^+$. Покажем, что парное решение систем (11) идентично "внешнему" оцениванию множества решений системы (1). Для этого воспользуемся идеологией знаковой методики.

Рассмотрим пример, когда заданы средняя матрица измерений A размерностью (5 × 5) и соответствующая ей матрица относительных погрешностей ε_{ij}. Тогда, согласно (8), знаковые матрицы

$\mathrm{sgn}(\varepsilon^-)$ и $\mathrm{sgn}(\varepsilon^+)$ и матрицы A^- и A^+ могут быть представлены системами (12) и (13):

$$\overline{A} = \begin{pmatrix} 0{,}3968 & 0{,}7904 & 0{,}8335 & 0{,}5144 & 0{,}4070 \\ 0{,}8085 & 0{,}9493 & 0{,}7689 & 0{,}8843 & 0{,}7487 \\ 0{,}7551 & 0{,}3276 & 0{,}1673 & 0{,}5880 & 0{,}8256 \\ 0{,}3774 & 0{,}6713 & 0{,}8620 & 0{,}1548 & 0{,}7900 \\ 0{,}2160 & 0{,}4386 & 0{,}9899 & 0{,}1999 & 0{,}3185 \end{pmatrix};$$

$$\varepsilon = \begin{pmatrix} 0{,}0107 & 0{,}0099 & 0{,}0170 & 0{,}0163 & 0{,}0123 \\ 0{,}0018 & 0{,}0038 & 0{,}0112 & 0{,}0176 & 0{,}0198 \\ 0{,}0022 & 0{,}0099 & 0{,}0186 & 0{,}0198 & 0{,}0106 \\ 0{,}0027 & 0{,}0030 & 0{,}0139 & 0{,}0000 & 0{,}0096 \\ 0{,}0136 & 0{,}0011 & 0{,}0117 & 0{,}0173 & 0{,}0160 \end{pmatrix}.$$

Определенные по формуле (4) знаки компонент погрешности ε, влияющие на максимальные погрешности решения, можно записать как

$$\mathrm{sgn}(\varepsilon^+) = \begin{pmatrix} + & + & + & - & - \\ - & - & + & + & + \\ + & + & - & - & - \\ - & - & + & + & + \\ - & + & - & + & + \end{pmatrix}, \quad \mathrm{sgn}(\varepsilon^-) = \begin{pmatrix} - & - & - & + & + \\ + & + & - & - & - \\ - & - & + & + & + \\ + & + & - & - & - \\ + & - & + & - & - \end{pmatrix}. \quad (12)$$

В соответствии со знаками компонент матриц (11) коэффициенты граничных матриц A^- и A^+ примут вид

$$A^- = \begin{Vmatrix} a_{11} + \varepsilon_{11} & a_{12} + \varepsilon_{12} & \dots & a_{15} - \varepsilon_{15} \\ a_{21} - \varepsilon_{21} & a_{22} - \varepsilon_{22} & \dots & a_{25} + \varepsilon_{25} \\ \dots & \dots & \dots & \dots \\ a_{51} - \varepsilon_{51} & a_{52} + \varepsilon_{52} & \dots & a_{55} + \varepsilon_{55} \end{Vmatrix};$$

$$A^+ = \begin{Vmatrix} a_{11} - \varepsilon_{11} & a_{12} - \varepsilon_{12} & \dots & a_{15} + \varepsilon_{15} \\ a_{21} + \varepsilon_{21} & a_{22} + \varepsilon_{12} & \dots & a_{25} - \varepsilon_{25} \\ \dots & \dots & \dots & \dots \\ a_{51} + \varepsilon_{51} & a_{52} - \varepsilon_{52} & \dots & a_{55} - \varepsilon_{55} \end{Vmatrix}. \quad (13)$$

Как следует из предыдущего анализа, определитель матрицы A^- максимально уменьшится по сравнению с невозмущенной матрицей, а матрицы A^+ — увеличится. Получить переборкой такой результат довольно проблематично, учитывая, что число переборов знаков для погрешностей коэффициентов матрицы резко возрастает с ростом ее размерности как 2^{n^2}, так что для приведенного примера системы с матрицей A размером (5×5) для получения оптимального сочетания знаков относительных погрешностей ε_{ij} следовало бы проверить $2^{25} > 10^7$ вариантов. При применении "знаковой" методики выбор приращений коэффициентов матрицы и компонент правых частей, определяющих наибольшую погрешность решения, проводится простым анализом произведений (8).

Решения систем (11) с компонентами матриц и правых частей (10) определяют два граничных вектора неизвестных x^- и x^+ с компонентами

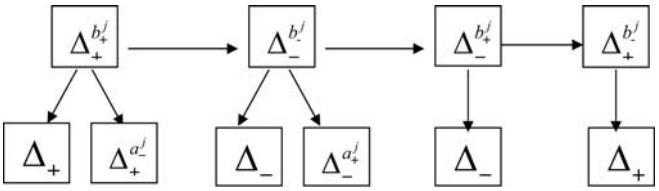
$$x_j^+ \in [x_j + x_j\varepsilon_{1j})], \; x_j^- \in [x_j - x_j\varepsilon_{2j})], \quad \forall j = \overline{1, n}, \; x = \{x^+, x^-\}, \quad (14)$$

где ε_{1j}, ε_{2j} — относительные погрешности вектора неизвестных.

Для получения верхних оценок решения (14) следует провести анализ исходя из вариабельности матриц при подстановках столбцов элементов и правых частей (9). Реальную наглядность этот метод приобретает при вычислении неизвестных по формулам Крамера. Так, для невозмущенной матрицы j-я компонента равна

$$x^j = \frac{\Delta^{b^j}}{\Delta}, \; \forall i = \overline{1, n}, \; \forall j = \overline{1, n}. \quad (15)$$

Подстановка столбцов правых частей b_-^j и b_+^j, компоненты которых имеют оптимальные приращения (9), в определитель Δ матрицы A определяет решения системы с отклонениями, зависящими только от погрешности вариаций правых частей системы. Подстановка вектора b^j в оптимально проварьированные по формулам (9) матрицы A^- и A^+ определяет компоненты вектора неизвестных в зависимости от возмущения коэффициентов матрицы. Следует отметить одно важное обстоятельство. Подстановка возмущенного вектора (имеется в виду уже получившего определенное знаковое приращение) в возмущенную матрицу определителя еще не определяет предельного приращения компоненты вектора неизвестных. Наибольший эффект может быть достигнут, когда определители числителя и знаменателя также имеют противоположные приращения. В этом случае мы имеем возможность рассчитывать на максимальное изменение компоненты вектора решения. Так например, компонента $x^j = \dfrac{\Delta_-^{b_-^j}}{\Delta_-^{a_+^j}}$ получается при замене b_-^j в числителе j-го столбца матрицы определителя Δ_-, полученного при максимальном уменьшении, а в знаменателе — заменой j-го столбца той же матрицы столбцом a_+^j при вычислении определителя, который был получен в сторону возрастания. Понятно, что значение j-й компоненты оказывается значительно меньшим, чем вычисленное по формуле (15). В другом варианте, когда j-й столбец матрицы A^+ определителя числителя Δ_+ заменить столбцом b_+^j, а в знаменателе аналогичный столбец заменить столбцом a_-^j, вычисленное значение компоненты $x^j = \dfrac{\Delta_+^{b_+^j}}{\Delta_+^{a_-^j}}$ получится значительно большим, чем по формуле (15) для невозмущенной системы.



На рисунке представлены шесть вариантов получения компонент возмущенных векторов, наиболее подозрительных на максимальные погрешности решения системы В узлах схемы размещены символы определителей: в верхних — числителя, в нижних — знаменателя. Вертикальные стрелки между узлами символизируют частное от деления числителя на знаменатель, указывая на определенный вариант вычисления одной и той же j-й компоненты x. Важно отметить, что как следует из формулы (14), вычисление максимальной или минимальной компоненты не влечет за собой таких же оптимальных значений других компонент. Поэтому по формулам Крамера (или соответственно через обратные матрицы) мы получаем только угловые точки векторов решения. Чтобы получить оценку разброса всех компонент решения, следует провести подстановку исходного и всех проварьированных столбцов векторов правых частей в исходную и проварьированные матрицы. Горизонтальные стрелки определяют переборку компонент вычисляемых векторов, среди которых определяются наибольшие и наименьшие по величине:

$$x_j^- = \min\{x_1^j, x_2^j, ..., x_k^j\}; \; x_j^+ = \max\{x_1^j, x_2^j, ..., x_k^j\},$$
$$j = \overline{1, n}, \; k = 6,$$

где x_j — j-я компонента, полученная в одном из шести вариантов расчета. Выборка максимального и минимального значений по каждой компоненте приводит к системе из двух граничных векторов

$$x^- = \{x_1^-, x_2^-, ..., x_n^-\}; \; x^+ = \{x_1^+, x_2^+, ..., x_n^+\},$$

определяющих решение задачи внешнего покоординатного оценивания или нахождения наиболее точных оценок решения интервальной системы для $\min\{x_v \in \Xi(\mathbf{A}, \mathbf{b})\}$ снизу и для $\max\{x_v \in \Xi(\mathbf{A}, \mathbf{b})\}$, $v = 1, 2, ..., k$, сверху для множества решений $\mathbf{X} = \{x \in \mathbf{R^n} | Ax = b, A^- \in \mathbf{A}, b^- \in \mathbf{b}, A^+ \in \mathbf{A}, b^+ \in \mathbf{b}\}$. Это равносильно отысканию для объемлющего прямоугольного параллелепипеда (так называемого бруса) сторон, представляющих компоненты x_j^-, x_j^+ векторов x^- и x^+ и решает задачу определения внешнего интервала для множества решений интервальной линейной системы (1).

По программе, формализованной отдельным m-файлом в системе МАТЛАБ, реализующей идеологию вышеописанного алгоритма, был проведен расчет погрешностей решения систем уравнений различного порядка, начиная со 2-го, 3-го, до размерности матриц 30 × 30 [8, 9]. Расчеты показали, что задаваемая погрешность исходных данных, как правило, не превосходящая 10 %, инициирует погрешность решения соответственно от нескольких до десятков процентов. С увеличением размерности матрицы системы вероятность увеличения погрешности возрастает в десятки раз, и во многом эта оценка зависит от вида коэффициентной системы уравнений. Конечно, это не значит, что такая погрешность может легко реализоваться, так как ее вероятность чрезвычайно мала в соответствии с определенным знаковым отклонением всех коэффициентов от измеряемых величин, но теоретически она возможна. Кроме того, если даже не все коэффициенты системы, а только какая-то часть из них получат соответствующее знаковое приращение, это уже может привести к значительному росту погрешности решения по сравнению с произвольно задаваемыми погрешностями или погрешностями одного знака. Поэтому именно знаковая оценка двусторонней погрешности коэффициентов линейной алгебраической системы может оказаться полезной при определении чувствительности ИСЛАУ как решения "внешней" задачи.

С этой целью по "знаковой" методике были разработаны алгоритм и программа оценки интервального решения ИСЛАУ, основанные на преобразования интервальной матрицы и правой части системы в две системы с точечными матрицами. Для расчета по программе задаются размерность системы, выбирается вид исследуемой матрицы (случайный или задаваемый заранее). В четырех внешних файлах содержатся граничные значения интервальных матриц и их правых частей. В начале вычислений проводится анализ определения невырожденности исходной и проварьированных матриц в целях достоверности дальнейших оценок. Относительные погрешности элементов матриц и векторы правых частей исходной системы вычисляются внутри программы. На экране выходного листинга фиксируются покомпонентные значения вектора решения системы со средней матрицей и систем с граничными матрицами с выборкой компонент из множества решений, имеющих максимально возможные отклонения в положительном и отрицательном направлениях, представляющих собой угловые значения. Система порядка $N = 10$ с относительной погрешностью 20 % считается в течение долей секунды. Увеличение размерности вводимой системы увеличивает время счета до нескольких минут и определяется скоростью вычисления обратных матриц, участвующих в реализации алгоритма. Полученные оценки решения интервальных систем "знаковым" и интервальными методами демонстрируют их полную совместимость и идентичность в пределах разбросов решений приводимых методов.

Рассмотрим в качестве примера параллельные оценки допустимого множества решений интервальных систем, полученных известными интервальными методами, и решений, полученных по "знаковой" методике точечных систем, предварительно преобразованных из интервальных.

Воспользуемся заданием одной и той же системы с интервальной матрицей размерностью 3×3 с различными правыми частями, взятой из литературных источников, и сравним приводимые там интервальные решения с полученными нами оценками по "знаковой" методике.

Пример 1. Интервальное матричное уравнение содержит матрицу, состоящую из интервальных коэффициентов и интервальных компонент правой части:

$$\begin{pmatrix} [3.7,\ 4.3] & [-1.5,\ -0.5] & [0,\ 0] \\ [-1.5,\ -0.5] & [3.7,\ 4.3] & [-1.5,\ -0.5] \\ [0,\ 0] & [-1.5,\ -0.5] & [3.7,\ 4.3] \end{pmatrix} \cdot \mathbf{x} = \begin{pmatrix} [-14;\ 14] \\ [-9;\ 9] \\ [-3;\ 3] \end{pmatrix}. \quad (16)$$

Значения интервалов для вектора решения $\mathbf{x}$, полученные методом Гаусса и по "знаковой" методике, совпадают для системы (16) с точностью до приведенного после запятой знака:

$$\mathbf{x} = \begin{pmatrix} [-6.38;\ 6.38] \\ [-6.40;\ 6.40] \\ [-3.40;\ 3.40] \end{pmatrix}.$$

Отметим, что в этом примере вектор правых частей содержит симметричные пределы, среднее значение каждого из которых равно 0.

Пример 2. Заменим в (15) вектор правых частей на вектор $\mathbf{b}^{\mathrm{T}} = ([2;\ 14],\ [3;\ 9],\ [-3;\ 1])$, оставив исходную матрицу системы:

$$\begin{pmatrix} [3.7,\ 4.3] & [-1.5,\ -0.5] & [0,\ 0] \\ [-1.5,\ -0.5] & [3.7,\ 4.3] & [-1.5,\ -0.5] \\ [0,\ 0] & [-1.5,\ -0.5] & [3.7,\ 4.3] \end{pmatrix} \cdot \mathbf{x} = \begin{pmatrix} [2;\ 14] \\ [3;\ 9] \\ [-3;\ 1] \end{pmatrix}.$$

Для этой системы в литературе приводятся результаты решения методом Гаусса, Хансена—Блика—Рона и методом Ньютона [3—5]. В последнем столбце (17) содержится покоординатная оценка решения по "знаковой" методике:

$$\begin{pmatrix} [0.517;\ 6.25] \\ [0.450;\ 6.07] \\ [-0.881;\ 2.73] \end{pmatrix}, \begin{pmatrix} [-0.206;\ 6.25] \\ [-0.386;\ 6.07] \\ [-2.01;\ 2.73] \end{pmatrix}, \begin{pmatrix} [0.523;\ 6.25] \\ [0.499;\ 6.07] \\ [-0.743;\ 2.73] \end{pmatrix}, \begin{pmatrix} [0.545;\ 6.25] \\ [0.666;\ 6.07] \\ [-0.743;\ 2.73] \end{pmatrix}. \quad (17)$$

Пример 3. При замене вектора $\mathbf{b}$ в (16) вектором $\mathbf{b}^{\mathrm{T}} = ([2;\ 14],\ [-9;\ -3],\ [-3;\ 1])$ те же методики, включая предлагаемую нами (последний столбец), приводят к следующим результатам:

$$\begin{pmatrix} [-1.09;\ 4.29] \\ [-4.02;\ 1.29] \\ [-2.44;\ 0.773] \end{pmatrix}, \begin{pmatrix} [-0.995;\ 5.01] \\ [-4.64;\ 1.52] \\ [-2.69;\ 1.38] \end{pmatrix}, \begin{pmatrix} [-0.995;\ 4.29] \\ [-3.79;\ 1.24] \\ [-2.35;\ 0.773] \end{pmatrix}, \begin{pmatrix} [-0.995;\ 4.286] \\ [-3.787;\ 1.240] \\ [-2.346;\ 0.773] \end{pmatrix}.$$

Из приведенных решений системы (16) видна значительная зависимость решения, включая и предлагаемый алгебраический подход, от влияния правых частей при одной и той же исходной интервальной матрице.

Заключение

- Заявлен новый алгебраический подход (знаковая методика) решения классической "внешней" задачи для интервальной линейной системы, позволяющей найти "внешние" покоординатные оценки множества решений.
- Определены допущения для использования метода, включая невырожденность точечных матриц, составляющих граничные матрицы, задание относительных погрешностей входных параметров, меньших 100 %, с дополнительными ограничениями на точность оценок при использовании знакового подхода для решения систем с матрицами, имеющими нулевое среднее коэффициентов.
- На ряде примеров показано применение знаковой методики для определения вектора неизвестных точечных матриц и подтверждена идентичность оценки чувствительности интервальных линейных систем интервальными и предложенным методами.
- Новый подход для решения интервальной задачи вместе с реализующим его численным алгоритмом, обладает

  — хорошей адаптируемостью к конкретным интервальным линейным системам;

  — универсальностью для оценки решений широкого класса задач.
- Реализация вычислительных процедур, основанная на "знаковой" методике, обеспечивается специально разработанными программами, формализованными *m*-файлами в системе МАТЛАБ.

Список литературы

1. **Петров Ю. П.** Как обеспечить надежность решения систем уравнений. Л.: БХВ-СПб, 2009. 172 с.
2. **Шарый С. П.** Конечномерный интервальный анализ. Новосибирск: Изд-во "XYZ". 2007. 700 с.
3. **Шарый С. П.** Алгебраический подход во "внешней задаче" для интервальных линейных систем // Фундаментальная и прикладная математика. 2002. Т. 8. № 2. С. 567—610.
4. **Hansen R. E.** On linear algebraic equations with interval coefficients // Topics in Interval Analysis / Ed. E. Hansen. Oxford Univ. Press. 1969. P. 35—46.
5. **Rohn J.** A Handbook of Results on Interval Linear Problems. Prague: Czech Academy of Sciences, 2005. April 7. 76 p.
6. **Шарый С. П.** Об "испанской версии" формального подхода к внешнему оцениванию множества решений интервальных линейных систем // Вычислительные технологии. 2011. Т. 16, № 3. С. 100—133.
7. **Иванова К. Ф.** Оценка погрешности численного решения уравнений Пуассона под воздействием флуктуаций входных параметров в среде Matlab // Санкт-Петербург: ПИЯФ РАН. 2010. 34 с.
8. **Иванова К. Ф.** Свидетельство РФ о государственной регистрации программы для ЭВМ № 2011611641 "Программный комплекс оценки экстремальных значений погрешности выходных характеристик решения стационарных задач строительной механики, тепловых и электромагнитных процессов (ПКОПП)". 2011.
9. **Иванова К. Ф.** Свидетельство РФ о государственной регистрации программы для ЭВМ № 2011617669 "Программа для оценки погрешности целевой функции при решении задачи линейного программирования (PZLP)". 2011.

ЖУРНАЛ В ЖУРНАЛЕ

НЕЙРОСЕТЕВЫЕ ТЕХНОЛОГИИ

№ 9
СЕНТЯБРЬ
2012

УДК 004.8

В. Ю. Осипов, д-р техн. наук, проф.,
Федеральное государственное
бюджетное учреждение науки,
Санкт-Петербургский институт информатики
и автоматизации Российской академии наук,
e-mail: osipov_vasiliy@mail.ru

Метод настройки ассоциативной интеллектуальной системы на входные сигналы

Рассмотрен метод настройки ассоциативной интеллектуальной системы на входные сигналы с учетом текущей загрузки системы. Данный метод расширяет возможности по обработке информации. Приведены математическая формулировка и алгоритм решения задачи, а также результаты моделирования.

Ключевые слова: *ассоциативная интеллектуальная система, нейронная сеть, настройка, сигналы*

Введение

Расширение возможностей ассоциативных интеллектуальных систем (АИС) по обработке информации представляет большой научный и практический интерес. Под АИС понимается совокупность взаимосвязанных датчиков, нейронной сети — искусственного "мозга" — и исполнительных устройств, предназначенных для обработки информации и взаимодействия с внешним миром в соответствии с воспринимаемыми закономерными связями между отдельными сигналами и их элементами.

Важное свойство АИС — способность при воздействии на них входных сигналов извлекать из долговременной памяти связанную с этими сигналами информацию. Достигнуты несомненные успехи по вызову из памяти АИС статических образов. Хуже обстоит дело с извлечением динамических сигналов из открытых АИС [1—5]. Это АИС с перестраиваемой структурой, постоянно обучающиеся в процессе обработки входных сигналов. К ним относятся АИС, ориентированные на непрерывный анализ и прогнозирование событий, интеллектуальное управление различными машинами и системами в труднопредсказуемых условиях, обработку речи, другие творческие задачи.

Одними из таких АИС выступают системы на основе рекуррентных нейронных сетей (РНС) с управляемыми синапсами [6—8]. Для этих, как и для других АИС вопросы совершенствования извлечения динамических сигналов из долговременной памяти далеко не исчерпаны. Для извлечения из долговременной памяти АИС запомненных сигналов в нее вводят неполные их описания или связанные с ними другие данные. Однако при этом недостаточно внимания уделяют настройке АИС на входные сигналы [1—4, 9—11]. Для успешного извлечения из памяти АИС запомненных воздействий необходимо иметь соответствующий уровень связи с ними входных сигналов.

В общем случае даже одни и те же по содержанию сигналы, вводимые в АИС, могут существенно отличаться друг от друга по форме (например, по значениям пространственных или временных характеристик, не несущих содержательной информации в конкретном случае). В АИС можно вводить динамические изображения удаляющихся, приближающихся, сдвигающихся, поворачивающихся, изменяющихся по цвету и яркости объектов, сигналы, соответствующие тихой и громкой, медленной и быстрой речи с различными паузами, и другие. Каждому из них свойственен свой уровень связи с запомненными сетью сигналами.

Для повышения уровня связи входных сигналов с запомненными ранее воздействиями сигналы преобразуют к виду, исключающему зависимость их от возможного перемещения, ротации и масштабирования [1—4, 9—11]. Этого добиваются как путем аналоговых, так и цифровых их преобразований. В частности, применяют быстрое преобразование Фурье, вейвлет (волновое) преобразование, аффинное и другие. Изменяют также энергетические и частотные характеристики сигналов. Во всех случаях сигнал перед введением в РНС АИС раскладывают на составляющие.

К недостаткам известных методов настройки АИС на входные сигналы относят: несовершенство используемых показателей и критериев эффективности настройки; слабый учет динамики изменения результатов ассоциативного вызова информации из долговременной памяти РНС, текущей загрузки оперативной памяти сети; не проработаны вопросы очередности параметров, по которым должна осуществляться рассматриваемая настройка. Согласно известным методам АИС лишены возможности самостоятельно настраиваться на входные сигналы по результатам ассоциативного вызова информации из долговременной памяти РНС.

Несовершенство известных методов настройки АИС на входные сигналы существенно ограничивает их возможности по интеллектуальной обработке информации.

Предлагается метод настройки АИС на входные сигналы по результатам ассоциативного вызова информации из ее долговременной памяти с учетом текущей загрузки системы.

Структура АИС и постановка задачи

Задана ассоциативная интеллектуальная система со структурой, приведенной на рис. 1. На вход АИС в общем случае можно подавать динамические сигналы различной физической природы, например, световой поток от наблюдаемых объектов, акустические сигналы (речь, музыка и другие звуки). После прямого преобразования в соответствующем блоке АИС они поступают в рекуррентную нейронную сеть (РНС) с управляемыми синапсами [6—8] в виде последовательных совокупностей единичных образов, несущих информацию о пространственных, частотных, амплитудных и фазовых характеристиках входных сигналов. Для формирования таких последовательностей входные сигналы раскладываются, в общем случае, на пространственно-частотные составляющие. Каждая составляющая преобразуется в последовательность единичных образов с частотой следования как функции от амплитуды этой составляющей [12]. Для настройки АИС на входные сигналы в блоке прямого преобразования (см. рис. 1) до разложения на составляющие изменяется их уровень, а также параметры перемещения, ротации и пространственного масштабирования.

При передаче совокупностей единичных образов от слоя к слою в РНС путем управления ее синапсами осуществляются пространственные сдвиги этих совокупностей. За счет таких сдвигов каждый слой сети разбивается на логические поля, а сами совокупности единичных образов продвигаются вдоль слоев в заданных направлениях. Сеть наделяется логической структурой. Варианты логических структур такой РНС могут быть различными: линейная, спиральная и другие [13]. Один из примеров логической структуры РНС приведен на рис. 2 *а*, *б*, *в*. На рис. 2, *а* показан вид сверху на первый слой сети, а на рис. 2, *б*, *в* — поперечные "срезы" вдоль, соответственно, первой и второй строк по двум слоям. Второй слой идентичен первому. Первый слой РНС, как и второй, в примере логически разбит на три строки по 25 полей в каждой. Ввод информации осуществляется через первое поле первого слоя, а вывод — с последнего поля второго слоя. Введенные в сеть последовательные совокупности единичных образов, продвигаемые вдоль слоев сети, на логическом уровне представляют собой буквы и составляют слово "сигнал". Каждой такой букве поставлена в соответствие длительность наблюдения и частоты следования, свойственных ей единичных образов. На рис. 2, *а* для всех букв они одинаковые. Информация о каждой букве передается в сети элементарной последовательностью, состоящей из пяти одинаковых совокупностей единичных образов.

Совокупности единичных образов первой и третьей строк (рис. 2, *б*, *в*) синхронно продвигаются вдоль слоев сети слева направо, а совокупности второй строки — навстречу им. Направления продвижения совокупностей показаны на рис. 2 стрелками. Такое встречное продвижение позволяет обеспечивать в сети широкое ассоциативное взаимодействие совокупностей единичных образов друг с другом.

При продвижении вдоль слоев каждая совокупность связывается с ближайшими к ней совокупностями. Чем ближе совокупности друг к другу, тем сильнее сила их связей. Совместно они также вы-

Рис. 1. Ассоциативная интеллектуальная система

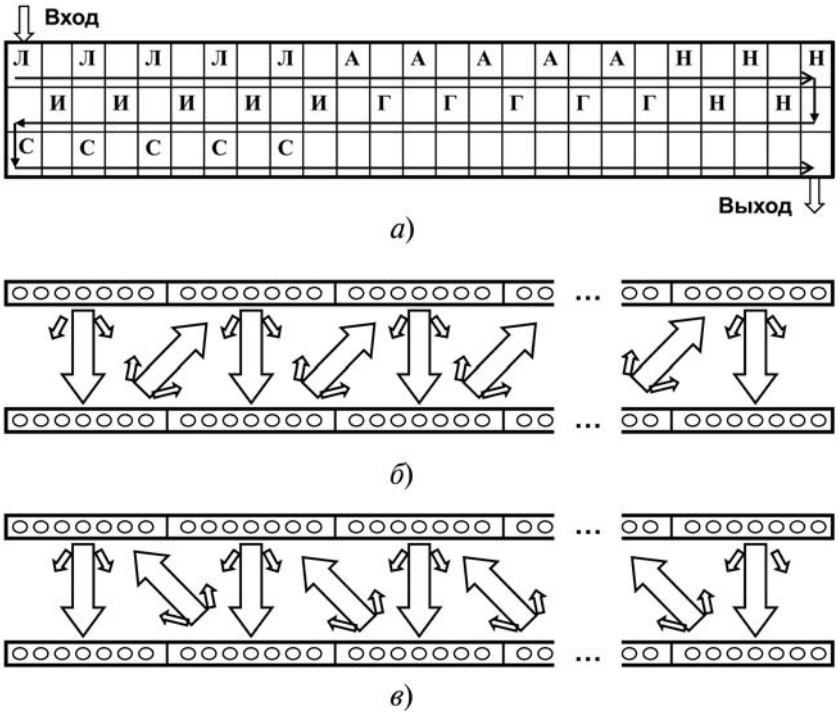


Рис. 2. Спиральная структура рекуррентной нейронной сети

зывают из долговременной памяти РНС ассоциированные с ними сигналы. Связывание совокупностей единичных образов друг с другом проявляется в изменении весов синапсов нейронов первого и второго слоев сети, а извлечение связанной информации из долговременной памяти — через возбуждение соответствующих нейронов. Долговременной памятью обладают синапсы, их веса по абсолютной величине не превышают единицы. Оперативная память — это память на самих нейронах, которые могут находиться в состояниях ожидания, возбуждения и временной невосприимчивости.

Для настройки АИС на временные и фазовые характеристики входных сигналов в ней могут изменяться задержки совокупностей единичных образов при передаче от одного слоя к другому [8].

Вне зависимости, по какому параметру осуществляется настройка АИС на входные сигналы, в конечном итоге она сводится к изменению пространственно-временной структуры сигналов в плоскости слоев РНС. Например, за счет варьирования отмеченными временными задержками можно растягивать или сжимать последовательности совокупностей единичных образов плоскости слоев, изменять расстояние между совокупностями и их взаимное расположение. При усилении или ослаблении входных сигналов изменяются частоты следования совокупностей в плоскости слоев. В случаях сдвигов сигналов, их поворотов, варьирования пространственным масштабом изменяется взаимное положение единичных образов в совокупностях на плоскости слоев.

В интересах недопущения перегрузки оперативной памяти РНС АИС можно также управлять порогом возбуждения ее нейронов без нарушения пространственно-временной структуры входных сигналов. Однако при этом существенно изменяются возможности по ассоциативному вызову из долговременной памяти связанной информации.

В РНС АИС за счет приоритетности коротких связей устанавливается однозначное соответствие между выходом и входом. Информация о значениях частот составляющих, на которые раскладывается входной сигнал, закрепляется за номерами соответствующих нейронов. При таких условиях по последовательностям совокупностей единичных образов на выходе РНС можно восстановить соответствующие им исходные сигналы [12]. Такое восстановление осуществляется в блоке обратного преобразования сигналов АИС (см. рис. 1).

Возможны ситуации, когда продвигающиеся по РНС совокупности единичных образов имеют низкий уровень извлечения из долговременной памяти связанных сигналов. Это может быть обусловлено следующими причинами:

- входные совокупности единичных образов являются относительно новыми и слабо связаны с ранее запомненными сигналами;
- свободное пространство оперативной памяти РНС исчерпано, дальнейшее извлечение сигналов из долговременной памяти в оперативную может привести к перегрузке РНС;
- введенные в РНС совокупности единичных образов не согласованы по параметрам с запомненными в долговременной памяти сигналами.

Чем быстрее АИС настроится на входные сигналы, тем раньше начнется полноценный ассоциативный вызов запомненной информации.

В интересах расширения функциональных возможностей АИС требуется разработать метод ее настройки на входные сигналы по результатам ассоциативного вызова информации из долговременной памяти с учетом текущей загрузки системы.

Математическая формулировка задачи

С математической точки зрения эту задачу можно сформулировать в следующем виде. Требуется на заданном интервале времени относительно момента t_k найти целесообразные i-е параметры $u^k_{ir\_opt}$ настройки АИС на входные сигналы на r-х тактах ее функционирования. При этих параметрах должен достигаться максимум числа ассоциативно вызываемых из долговременной памяти РНС единичных образов,

$$F^k_o = \max_j \sum_{r=1}^{T} F^k_{jr}(u^k_{ijr}), \tag{1}$$

при условиях:

$$F^k_{jr} = F^k_{jr\_1} + F^k_{jr\_2}; \tag{2}$$

$$F^k_{jr\_1} = N^k_{jr\_2} - N^k_{jr-1\_1} \leqslant E(N^k_{jr-1\_1}); \tag{3}$$

$$F^k_{jr\_2} = N^k_{jr\_1} - S^k_{r-1\_\text{вх}} - N^k_{jr-1\_2} + S^k_{r-1\_\text{вых}} \leqslant E(N^k_{jr-1\_2}); \tag{4}$$

$$u_{1i} \leqslant u^k_{ijr} < u_{2i}; \tag{5}$$

$$i = \overline{1, I}\ ;\ j = \overline{1, J}\ ;\ r = \overline{1, T}\ .$$

В формулах (1)—(5) приняты следующие обозначения: $F^k_{jr}(\cdot)$ — число ассоциативно вызываемых из долговременной памяти РНС единичных образов на r-м такте (шаге) ее функционирования при j-м варианте значений i-х параметров u^k_{ijr} настройки относительно момента t_k; I — число всех настраиваемых параметров АИС; T — интервал времени, на котором осуществляется настройка относительно момента t_k; $F^k_{jr\_1}$, $F^k_{jr\_2}$ — число ассоциативно вызываемых, соответственно, на первом и втором слоях единичных образов; $N^k_{jr-1\_1}$, $N^k_{jr\_1}$ —

число единичных образов на первом слое РНС на $(r-1)$-м и r-м тактах; $N^k_{jr-1\_2}$, $N^k_{jr\_2}$ — число единичных образов на втором слое на $(r-1)$-м и r-м тактах; $S^k_{r-1\_\text{вх}}$ — число единичных образов во входной совокупности, вводимой в РНС через первый слой на $(r-1)$-м такте; $S^k_{r-1\_\text{вых}}$ — число единичных образов в выходной совокупности на втором слое, покидающей сеть на $(r-1)$-м такте; $E(N^k_{jr-1\_1})$, $E(N^k_{jr-1\_2})$ — допустимое число ассоциативно вызываемых, соответственно, на втором и первом слоях единичных образов в зависимости от текущей загрузки передающих слоев; u_{1i}, u_{2i} — нижняя и верхняя границы допустимых значений настраиваемых параметров.

Согласно условиям (3), (4) на каждом слое РНС в одном такте ее функционирования не может быть ассоциативно вызвано более чем $E(N^k_{jr-1\_1})$ или $E(N^k_{jr-1\_2})$ единичных образов. В противном случае сеть перегрузится. Через функции $E(N^k_{jr-1\_1})$, $E(N^k_{jr-1\_2})$ осуществляется учет текущей загрузки РНС при настройке АИС на входные сигналы.

В связи с тем, что пространство возможных значений настраиваемых параметров АИС велико, а время ограничено, для решения задачи (1)—(5) необходима разработка специальных алгоритмов. Заметим, что входные сигналы должны восприниматься АИС в широком диапазоне их параметров.

Алгоритм настройки АИС на входные сигналы

В этих условиях одновременный поиск всех целесообразных параметров настройки АИС на входные сигналы представляет чрезвычайно трудную задачу. Однако ситуация облегчается учетом ряда особенностей такой настройки, позволяющих ее упростить, разбить на подзадачи по числу видов управляемых параметров и решать их последовательно. При этом открытым остается вопрос, насколько и в какой последовательности нужно изменять параметры АИС, чтобы быстрее настроиться на входные сигналы. Для определения последовательности изменения параметров настройки АИС на входные сигналы можно воспользоваться принципом аналогии с биологическими системами. Полагается, что человек, наблюдая за объектами, сначала настраивается своим зрением на световой поток от окружающего фона, затем фокусируется на объекте и исследует его по частям, переводя взгляд с одной точки на другую, при необходимости наклоняет голову влево или вправо, настраивается на динамику изменения параметров объекта. При потере интереса к объекту переводит взгляд с одного направления наблюдения на другое, поворачивает голову, сдвигается в пространстве, настраивается на новый фон и фокусируется на новом объекте. Эта схема с позиции общих положений справедлива и для восприятия акустических сигналов.

Принимая это во внимание, с использованием принципа наискорейшего спуска, предлагается следующий алгоритм настройки АИС на входные сигналы, обеспечивающий достижение максимума показателя (1).

Шаг 1. Поиск $F^k_{o1} = \max\limits_j \sum\limits_{r=1}^{T_1} F^k_{jr}(u^k_{1jr})$ на заданном интервале времени $T_1 - 1$ путем варьирования амплитудой и, соответственно, энергией входных сигналов через изменение их коэффициента усиления u^k_{1jr} с учетом условий (2)—(5). В результате реализации данного шага определяется целесообразное значение коэффициента усиления $u^k_{1r\_opt}$.

Шаг 2. Поиск $F^k_{o2} = \max\limits_j \sum\limits_{r=T_1}^{T_2} F^k_{jr}(u^k_{2jr}, u^k_{3jr})$ на интервале времени $T_2 - T_1$ путем варьирования параметрами u^k_{2jr}, u^k_{3jr} сжатия (расширения) сигналов по осям *X, Y* в плоскости *XY* наблюдения с учетом (2)—(5). Получение целесообразных значений параметров сжатия (расширения): $u^k_{2r\_opt}$, $u^k_{3r\_opt}$.

Шаг 3. Поиск $F^k_{o3} = \max\limits_j \sum\limits_{r=T_2}^{T_3} F^k_{jr}(u^k_{4jr}, u^k_{5jr})$ на интервале $T_3 - T_2$ путем варьирования параметрами u^k_{4jr}, u^k_{5jr} сдвигов сигналов по осям *X, Y* с учетом (2)—(5). Выбор целесообразных значений параметров сдвигов: $u^k_{4r\_opt}$, $u^k_{5r\_opt}$.

Шаг 4. Поиск $F^k_{o4} = \max\limits_j \sum\limits_{r=T_3}^{T_4} F^k_{jr}(u^k_{6jr})$ на интервале $T_4 - T_3$ путем варьирования углом u^k_{6jr} поворота сигналов с учетом (2)—(5). Определение целесообразного угла поворота: $u^k_{6r\_opt}$.

Шаг 5. Поиск $F^k_{o5} = \max\limits_j \sum\limits_{r=T_4}^{T_5} F^k_{jr}(u^k_{7jr})$ на интервале $T_5 - T_4$ путем варьирования временем задержки u^k_{7jr} совокупностей единичных образов, передаваемых от слоя к слою нейронной сети,

с учетом (2)—(5). Получение целесообразного значения задержки: $u^k_{7r\_opt}$.

Шаг 6. $T = T_1 + T_2 + T_3 + T_4 + T_5$. $F^k_o = F^k_{o1} + F^k_{o2} + F^k_{o3} + F^k_{o4} + F^k_{o5}$. Если процесс функционирования АИС не прерывается, то $k = k + 1$ и переход к шагу 1. При этом, чем больше текущее значение F^k_{os}, тем меньше пределы варьирования рассматриваемыми параметрами на следующем шаге.

Согласно этому алгоритму при наличии рассогласования сигналов по какому-либо параметру должно осуществляться варьирование им. Это позволяет находить не только направления изменения параметров, повышающие F^k_o, но и новые внешние воздействия, стимулирующие ассоциативное запоминание и извлечение информации из долговременной памяти РНС.

Для оперативного достижения локального оптимума F^k_{o1} коэффициент усиления следует увеличивать, если текущий уровень входного сигнала ниже некоторого предварительно заданного среднего значения, и уменьшать, когда он выше. При оперативной настройке АИС по параметрам пространственного сжатия (расширения) входных сигналов принцип наискорейшего спуска проявляется в следующем: чем больше прирост показателя, тем ближе параметры настройки к оптимальным значениям. Этот же прием применим для быстрого выхода и на другие целесообразные параметры настройки АИС. При этом следует учитывать значение прироста показателей на каждом шаге функционирования РНС с учетом значений $E(N^k_{jr-1\_1})$, $E(N^k_{jr-1\_2})$. Удержание объекта в поле зрения АИС должно осуществляться только исходя из интереса к нему системы, проявляющегося в высоком уровне ассоциативного вызова из долговременной памяти РНС связанной с ним информации.

При соблюдении ряда условий варьирование параметрами входных сигналов может существенно не сказываться на точности обратного преобразования [12] совокупностей единичных образов в соответствующие им исходные сигналы. Это справедливо, например, когда каждому входному динамическому сигналу в сети соответствует значительное число совокупностей единичных образов и T существенно меньше длительности этого сигнала. В реальных биологических системах это условие всегда выполняется.

Чтобы реализовать ассоциативную интеллектуальную систему с высокими "мыслительными способностями", необходимо наличие вызова из долговременной памяти одних сигналов другими при длительном отсутствии внешних воздействий.

Одним из условий обеспечения такого вызова выступает соблюдение некоторого баланса между покидающими сеть совокупностями единичных образов и вызываемыми из долговременной памяти сигналами.

Результаты моделирования

В интересах подтверждения справедливости сформулированных положений по настройке АИС на входные сигналы была разработана программная модель ее двухслойной РНС с управляемыми синапсами. Для исключения перегрузки оперативной памяти РНС в ней осуществлялась регулировка порогом возбуждения нейронов в зависимости от текущей загрузки ее слоев единичными образами. С формальной точки зрения за такую регулировку в модели (1)—(5) отвечают функции $E(N^k_{jr-1\_1})$, $E(N^k_{jr-1\_2})$. Имитацию настройки АИС на входные сигналы осуществляли изменением их параметров. Изменяли частоту следования совокупностей единичных образов при заданной длительности сигналов и длительность сигналов при постоянстве их энергии. Осуществляли сжатие и расширение, а также сдвиги и повороты совокупностей единичных образов, вводимых в сеть, относительно эталонных сигналов. В интересах этого осуществлялись аффинные преобразования над входными совокупностями единичных образов. Каждому единичному образу входной совокупности с координатами x и y после аффинных преобразований ставились в соответствие координаты $x^* = \eta x\cos\varphi - \xi y\sin\varphi + b$, $y^* = \eta x\sin\varphi - \xi y\cos\varphi + d$, где η, ξ — параметры сжатия (расширения); φ — угол поворота; b, d — параметры сдвигов по осям X, Y.

Слои РНС содержали по 2100 нейронов. За счет пространственных сдвигов передаваемых совокупностей единичных образов от слоя к слою каждый слой логически разбивали на две строки, содержащие по 25 одинаковых полей, размером 6×7 нейронов. Сигналы в сеть вводили через первое поле, а снимали с последнего поля. Совокупности единичных образов продвигались вдоль слоев по спирали, как и на рис. 2, *а*. Для исключения перегрузки долговременной памяти РНС при запоминании входных сигналов осуществлялось частичное стирание с синапсов ранее запомненных результатов распознавания.

В результате моделирования процессов настройки АИС на входные сигналы и процессов вызова ими связанной информации из долговременной памяти РНС установлено следующее. За счет последовательной настройки АИС с сужением пределов варьирования ее параметрами с ростом числа ассоциативно вызываемых из долговременной памяти единичных образов можно быстро выйти на целесообразные параметры. Только при

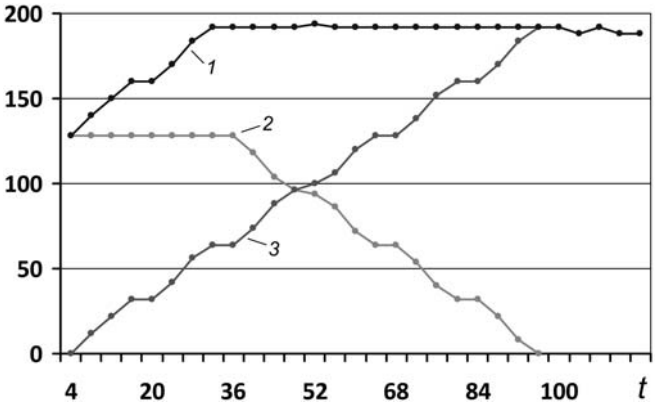


Рис. 3. Число единичных образов в оперативной памяти сети: *1* — всего; *2* — введенных через вход; *3* — ассоциативно вызванных из долговременной памяти

согласовании параметров входных воздействий с запомненными сигналами обеспечивается как максимум числа ассоциативно вызываемых из памяти единичных образов, так и точность воспроизведения запомненных динамических сигналов.

Для обеспечения высокой эффективности вызова из долговременной памяти РНС запомненной информации необходимо использовать короткие, но сильно связанные с ней сигналы — запросы и делать паузы между ними. Это объясняется тем, что при большой загрузке РНС возможности по ассоциативному вызову входными сигналами из ее долговременной памяти запомненной информации существенно ниже, чем при малой загрузке. Здесь сказывается роль регулировки порогом возбуждения нейронов в зависимости от текущей загрузки слоев сети единичными образами.

Установлено также, что при наличии согласования по всем параметрам вызывающих воздействий с извлекаемыми сигналами за счет регулирования порогом возбуждения нейронов можно реализовать устойчивый вызов из долговременной памяти сети одних сигналов другими. На рис. 3 приведены результаты одного из примеров такого вызова, где время указано в условных единицах. В соответствии с ними введенные в сеть единичные образы (кривая *2*), входящие в соответствующие совокупности, с течением времени *t* покидают ее, но при прохождении по сети они ассоциативно вызывают из долговременной памяти запомненные сигналы (кривая *3*). Затем уже вызванные из долговременной памяти сигналы вызывают следующие, связанные с ними. Число вызванных из долговременной памяти единичных образов в примере растет до тех пор, пока более ранние вызванные образы не начнут покидать сеть. Кривая *1* на рис. 3 соответствует числу всех единичных образов, находящихся в оперативной памяти сети.

Заключение

Предлагается метод настройки ассоциативной интеллектуальной системы на входные сигналы, предусматривающий последовательное варьирование соответствующими параметрами с учетом текущей загрузки ее нейронной сети. В качестве показателя этой настройки рекомендуется использовать число ассоциативно извлекаемых из долговременной памяти сети единичных образов.

За счет последовательной настройки АИС с сужением пределов варьирования ее параметрами с ростом числа ассоциативно вызываемых из долговременной памяти единичных образов можно быстро выходить на целесообразные параметры.

Только при настроенной системе на входные сигналы обеспечивается как максимум числа ассоциативно вызываемых из долговременной памяти единичных образов, так и точность воспроизведения запомненных ею динамических сигналов.

Для исключения перегрузки оперативной памяти РНС необходимо управлять порогом возбуждения нейронов сети, исходя из предположения, что вызывающие и извлекаемые из долговременной памяти сигналы согласованы по своим характеристикам.

Предлагаемый метод позволяет расширить возможности АИС по обработке информации и может быть использован при создании перспективных ассоциативных интеллектуальных систем.

Список литературы

1. **Galushkin A. I.** Neural Networks Theory. Berlin—Heidelberg: Springer-Verlag, 2007. 396 p.

2. **Хайкин С.** Нейронные сети: полный курс, 2-е издание.: Пер. с англ. М.: Вильямс, 2006. 1103 с.

3. **Осовский С.** Нейронные сети для обработки информации / Пер. с польского И. Д. Рудницкого. М.: Финансы и статистика, 2002. 344 с.

4. **Haikonen Pentti O. A.** The Role of Associative Processing in Cognitive Computing // Cognitive Computing. 2009. N 1. P. 42—49.

5. **Ivanscevic V. G., Ivanscevic T. T.** Neuro-Fuzzy Associative Machinery for Comprehensive Brain and Cognition Modelling. Berlin Heidelberg: Springer-Verlag, 2007. 720 p.

6. **Осипов В. Ю.** Рекуррентная нейронная сеть с управляемыми синапсами // Информационные технологии. 2010. № 7. С. 43—47.

7. **Осипов В. Ю.** Устойчивость рекуррентных нейронных сетей с управляемыми синапсами // Информационные технологии. 2011. № 9. С. 69—73.

8. **Осипов В. Ю.** Нейронная сеть с прошедшим, настоящим и будущим временем // Информационно-управляющие системы. 2011. № 4.

9. **Grossberg S., Huang Tsung-Ren.** ARTSCENE: A Neural System for Natural Scene Classification. Technical Report CAS/CNS-TR-07-017. Boston: Boston University, 2007. 28 p.

10. **Карпов В. Э., Вальцев В. Б.** Динамическое планирование поведения робота на основе сети "интеллектуальных" нейронов // Искусственный интеллект и принятие решений. 2009. № 2.

11. **Rosemarie Velik.** Why Machines Cannot Fell // Minds & Machines. 2010. N 20. P. 1—18.

12. **Осипов В. Ю.** Прямое и обратное преобразование сигналов в ассоциативных интеллектуальных машинах // Мехатроника, автоматизация, управление. 2010. № 7. С. 27—32.

13. **Осипов В. Ю.** Оптимизация ассоциативных интеллектуальных систем // Мехатроника, автоматизация, управление. 2011. № 3. С. 35—39.

УДК 621.395

Э. К. Алгазинов,
д-р физ.-мат. наук, проф., зав. каф.,
М. А. Дрюченко,
канд. техн. наук, ассистент каф.,
Е. Ю. Митрофанова, аспирант,
e-mail: mitrofanova@cs.vsu.ru,
А. А. Сирота, д-р техн. наук, проф.,
e-mail: sir@cs.vsu.ru,
Воронежский государственный университет

Математическое и программное обеспечение для создания цифровых водяных знаков с использованием искусственных нейронных сетей

Описываются алгоритмы обработки информации и реализованный на их основе программный комплекс, предназначенные для создания цифровых водяных знаков как средства защиты авторских прав на объекты цифрового контента. Основой для алгоритмов создания цифровых водяных знаков являются нейросетевые функциональные модели преобразования данных.

Ключевые слова: *цифровые водяные знаки, нейронные сети, цифровой контент, стеганография*

Введение

Проблема защиты мультимедиа объектов от подделки и копирования определяет необходимость использования новых информационных технологий. В этой области одной из перспективных является технология скрытного "маркирования" защищаемых файлов-контейнеров с помощью специальных невидимых для посторонних лиц меток — цифровых водяных знаков (ЦВЗ) [1—3]. Основной проблемой при реализации технологий создания ЦВЗ является сохранение качества маркируемых путем внедрения ЦВЗ файлов при их использовании по основному назначению в сочетании с надежностью дальнейшего восстановления ЦВЗ. Единого стандарта и универсальной технологии создания ЦВЗ не существует. Компании-разработчики предлагают различные варианты, начиная от фактически видимых ЦВЗ, закрывающих смысловую часть защищаемых данных, и заканчивая невидимыми цифровыми метками, для восстановления которых необходимы специальные программные средства. Для создания скрытых ЦВЗ используются методы компьютерной стеганографии [2, 3].

В настоящее время разработано достаточно много алгоритмов стеганографического скрытия информации (ССИ) в файлах-контейнерах различных форматов. Среди недостатков известных алгоритмов можно отметить их зависимость от формата контейнера, невысокую устойчивость к трансформациям маркированных данных, трудоемкость вычисления, необходимость наличия исходного файла для извлечения скрываемых данных, а также "алгоритмический" характер реализуемых преобразований. В определенной степени преодолеть подобные недостатки можно с использованием возможностей искусственных нейронных сетей (ИНС) [4—7]. В частности, в работах [6, 7] рассматривался подход к компьютерной стеганографии, основанный на реализации нейросетевых функциональных моделей преобразования данных, в рамках которого специально обученные нейронные сети используются для реализации скрывающего и восстанавливающих преобразований. Показано, что такой процесс встраивания данных в файл-контейнер носит существенно менее прозрачный характер, чем в большинстве известных алгоритмов стеганографии. Предложенные в работах [6, 7] алгоритмы встраивания ориентированы для реализации ССИ в файлах вещественных форматов данных и не являются эффективными для объектов, имеющих целочисленные форматы представления данных. Тем не менее, данный подход, основанный на использовании нейросетевых функциональных преобразований данных, может использоваться и при разработке новых технологий создания ЦВЗ. Поэтому представляет интерес его развитие для создания скрытых ЦВЗ в распространенных объектах цифрового контента, представленных как в вещественных, так и в целочисленных форматах (изображения в форматах *.bmp,*.jpeg и др., аудио- и видеоданные *.wav, *.avi, *.mp4 и др.). Таким образом, целью работы является разработка универсального математического и программного обеспечения, реализующего нейросетевые технологии создания ЦВЗ повышенной скрытности и устойчивости в объектах цифрового контента, представленных в файлах распространенных форматов.

Нейросетевые модели создания ЦВЗ

В общем виде задача создания ЦВЗ путем ССИ может быть сформулирована следующим образом. Пусть Z, D, K есть множество возможных контейнеров, множество скрываемых сообщений и множество ключей. Тогда процедура встраивания сообщений может быть представлена в виде отображения

$$F{:}Z \times D \times K \to \tilde{Z},\ \tilde{z} = F(z, d, k),$$
$$z \in Z,\ d \in D,\ k \in K,$$

где $\tilde{Z}$ — множество заполненных файлов-контейнеров. При этом следует обеспечить $\|z - \tilde{z}\| \to \min$ и $F(z, d, k) \approx F(z + \varepsilon, d, k)$, т. е. свойства контейнера должны модифицироваться так, чтобы внесенное при встраивании ЦВЗ изменение контейнера практически невозможно было бы выявить при ви-

зуальном и статистическом контроле, а сам ЦВЗ должен быть максимально устойчивым к различным преобразованиям. Соответствующая постановка задачи с использованием аппарата ИНС может быть сформулирована следующим образом. Требуется с использованием функциональных возможностей нейронных сетей для любого фрагмента контейнера, представленного в виде вектора $z \in R^n$, и вектора встраиваемых данных $d \in R^m$, $m \ll n$, построить отображения

$$\tilde{z} = F_1(z),\ \tilde{z} \in \tilde{Z},\ \bar{z} = F_2(\tilde{z}, d),\ \bar{z} \in \bar{Z},$$

$$\|\bar{z} - z\| \to \min,\ \tilde{d} = F_3(\bar{z}),\ \tilde{d} \in D,\ \|d - \tilde{d}\| \to \min,$$

где оператор F_1 реализует сжимающее отображение входных данных, обеспечивающее подготовку вектора-контейнера к встраиванию данных, оператор F_2 реализует собственно встраивание ЦВЗ, а оператор F_3 — восстановление ранее скрытой информации.

В общем случае для решения данной задачи можно использовать нейронные сети различных типов и архитектур, которые обеспечивают принципиальные возможности воспроизведения функциональных моделей преобразования данных в соответствии с приведенными общими соотношениями. Будем искать указанные отображения на основе нейронных сетей прямого распространения и рассмотрим сначала реализацию описанной выше общей модели встраивания данных для работы с данными вещественного формата. Для обоснования базового алгоритма создания ЦВЗ изначально используем стохастическое представление данных. В качестве ЦВЗ без ограничения общности будем рассматривать двоичную последовательность $d^{(p)}$, $p = \overline{1, P}$, при этом $d^{(p)} \in \{-1, +1\}$, $p = \overline{1, P}$, — скалярная величина, которая несет в себе один бит информации в пределах встраиваемого сообщения. Пусть $z = (z_1, ..., z_n)^{\mathrm{T}}$ — случайный вектор, представляющий фрагмент файла контейнера, совокупность реализаций которого $z^{(p)}$, $p = \overline{1, P}$, используется для встраивания элементов последовательности ЦВЗ и, соответственно, в качестве обучающей выборки — для построения нейронной сети F_1. Пусть для определенности математическое ожидание вектора z равно $M[z] = 0$, а матрица ковариации равна $M[zz^{\mathrm{T}}] = R_z$.

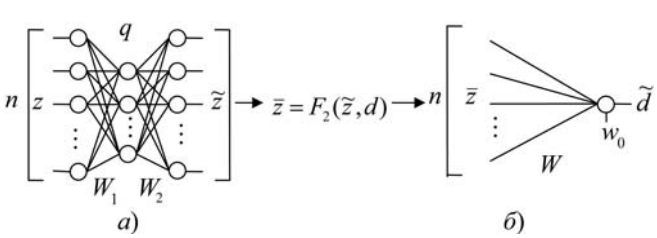


Рис. 1. Нейронная сеть, реализующая сжимающее преобразование для последующего встраивания (*а*); нейронная сеть, реализующая восстановление скрытой информации (*б*)

Структура линейной нейронной сети (НС), реализующей оператор F_1, в общем виде приведена на рис. 1, *а*. При этом число нейронов в скрытом слое нейронной сети равно $q \leqslant n$. Матрицы весовых коэффициентов первого и второго слоя сети обозначены соответственно W_1, W_2. Для реализации алгоритма встраивания данных сначала проводится обучение сети, обеспечивающей первое сжимающее отображение. Обучение проводится по совокупности реализаций входного и целевого вектора $D_{zy} = \{z^{(p),\,\mathrm{T}}, y^{(p)} = z^{(p)}\}$, $p = \overline{1, P}$, так чтобы минимизировать среднюю квадратичную ошибку представления входного вектора на выходе сети:

$$E = \frac{1}{2} \sum_{p=1}^{P} (y^{(p)} - W_2 W_1 z^{(p)})^{\mathrm{T}} (y^{(p)} - W_2 W_1 z^{(p)}) \to \min.$$

В результате обучения нейронной сети, представленной на рис. 1, *а*, получается индивидуальное для данного набора данных сжимающее отображение с весьма незначительными потерями, при котором реально получаемый на выходе сети вектор $\tilde{z} = (\tilde{z}_1, ..., \tilde{z}_n)^{\mathrm{T}}$ может быть представлен в виде разложения по первым q собственным векторам φ_i, $i = \overline{1, q}$, выборочной матрицы ковариации

$$\hat{R}_z^{(P)} = \frac{1}{P-1} \sum_{p=1}^{P} z^{(p)} z^{(p),\,\mathrm{T}},$$

которые являются одновременно функциями разложения Карунена—Лоэва [8, 9]. Тогда для любой реализации входного вектора $z^{(p)}$ получаемый на выходе вектор $\tilde{z}^{(p)}$ можно представить в виде

$$\tilde{z}^{(p)} = \sum_{i=1}^{q} \alpha_i^{(p)} \varphi_i,\ z^{(p)} = \sum_{i=1}^{q} \alpha_i^{(p)} \varphi_i,$$

где $\alpha_i^{(p)}$, $i = \overline{1, q}$, — коэффициенты разложения по первым $q \leqslant n$ собственным векторам φ_i, $i = \overline{1, q}$, матрицы $\hat{R}_z^{(P)}$.

В целях минимизации ошибки искажения контейнера при выполнении подобного разложения будем использовать сеть, для которой $q = n - 1$, тогда $\tilde{z}^{(p)}$ будет отличаться от $z^{(p)}$ только "высокочастотной" составляющей с малой амплитудой и дисперсией, соответствующей минимальному собственному числу выборочной матрицы ковариации.

После того как сеть F_1 обучена, выполняется оператор F_2, реализующий окончательное встраивание ЦВЗ в файл-контейнер. При этом вычисляется нормированный вектор

$$\varphi_n = r_{\min} / \sqrt{r_{\min}^{\mathrm{T}} r_{\min}},\ r_{\min} = \frac{1}{P} \sum_{p=1}^{P} (z^{(p)} - \tilde{z}^{(p)}),$$

и каждый фрагмент файла-контейнера, описываемый теперь (после сжатия) вектором $\tilde{z}^{(p)}$, $p = \overline{1, P}$,

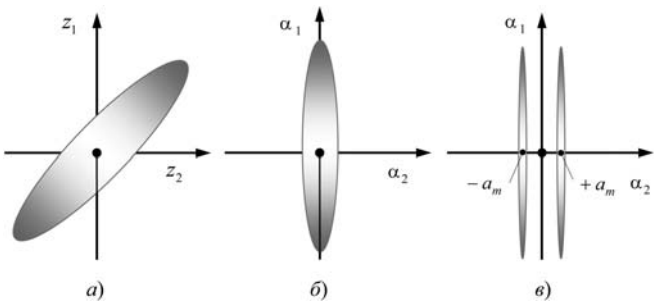


Рис. 2. Область локализации исходного случайного вектора (*а*), после выполнения декоррелирующего преобразования (*б*), после встраивания ЦВЗ (*в*)

модифицируется последовательностью ЦВЗ на основе следующего соотношения:

$$\bar{z}^{(p)} = \tilde{z}^{(p)} + a_m d^{(p)} \varphi_n,\ \tilde{z}^{(p)} = W_2 W_1 z^{(p)},\ p = \overline{1, P}\ , \quad (1)$$

где a_m — амплитуда вносимого искажения.

Соотношение (1) описывает результирующие действия, выполняемые при заполнении исходных фрагментов файла-контейнера, после чего образуется последовательность $\bar{z}^{(p)}$, $p = \overline{1, P}$. На рис. 2 дана наглядная иллюстрация описываемого процесса на примере преобразования данных, представляемых в виде гауссовского коррелированного вектора. На рис. 2, *а* показана исходная область локализации значений входного случайного вектора, которая имеет вид эллипсоида (гиперэллипсоида в многомерном пространстве). После выполнения декоррелирующего преобразования Карунена—Лоэва эллипсоид принимает вид, показанный на рис. 2, *б*. Рис. 2, *в* отражает вносимую модификацию входного вектора, когда вместо случайной величины $\alpha_n = \alpha_2$, распределенной по гауссовскому закону, используются два случайных точечных значения $a_2 = \pm a_m$, что соответствует использованию данных, имеющих две проекции гиперэллипсоида на ось α_1.

Для восстановления скрытых таким образом данных и, соответственно, для реализации оператора F_3 может быть использована нейронная сеть, архитектура которой показана на рис. 1, *б*. Такая сеть обучается для решения задачи классификации входного для нее вектора $\bar{z}$ в целях выделения значения ранее скрытого в нем элемента последовательности d. На рис. 1, *б* используются следующие обозначения: W — матрица весовых коэффициентов сети, w_0 — вектор постоянного смещения. Волнистая линия в обозначениях элементов восстанавливаемой последовательности означает наличие возможных ошибок при проведении подобной классификации последовательности входных сигналов $\bar{z}^{(p)}$, $p = \overline{1, P}$ в виде последовательности $\tilde{d}^{(p)}$, $p = \overline{1, P}$.

Для эффективного восстановления скрытых данных необходимо решить задачу классификации наблюдаемого вектора $\bar{z}$ по его принадлежности к одному из классов H_1 и H_2, характеризующихся различными математическими ожиданиями. В работах [6, 7] показано, что в рассматриваемом случае указанное решающее правило получается путем обучения простейшей однослойной линейной нейронной сети, структура которой приведена на рис. 1, *б*. При этом для гауссовских случайных векторов в результате обучения ИНС (рис. 1, *б*) по совокупности $\{\bar{z}^{(p)}, d^{(p)}, p = \overline{1, P}\}$ при $P \to \infty$ формируется преобразование, реализующее структуру оптимального решающего правила.

Для работы с файлами-контейнерами, имеющими целочисленный формат представления данных, необходимо определенным образом модифицировать описываемый выше базовый алгоритм. Проблема состоит с том, что данные, используемые для обучения нейронной сети, а также параметры сети должны быть заданы в вещественном формате, тогда как входные и выходные данные, а также данные, используемые для модификации контейнера при встраивании, имеют целочисленное представление. Поэтому требуется соблюдать определенный порядок преобразования данных в ходе выполнения процедур встраивания ЦВЗ.

Прежде всего любой исходный вектор $g^{(p)}$ (фрагмент контейнера), имеющий целочисленное представление, преобразуется в вектор $z^{(p)}$, содержащий вещественные данные в диапазоне значений −0,5...+0,5 (с центрированием)

$$z^{(p)} = (z_1^{(p)}, ..., z_n^{(p)})^{\mathrm{T}} = F_{double}(g_1^{(p)}, ..., g_n^{(p)}) - 0{,}5,$$

где $F_{double}(g_1^{(p)}, ..., g_n^{(p)})$ — функция преобразования целочисленных данных в данные вещественного формата в диапазоне 0, ..., 1. Далее в соответствии с приведенным выше алгоритмом проводится обучение нейронной сети F_1 по выборке $z^{(p)}$, $p = \overline{1, P}$.

После обучения входная последовательность данных $z^{(p)}$, $p = \overline{1, P}$, пропускается через сеть и формирует результирующую последовательность $\tilde{z}^{(p)}$, $p = \overline{1, P}$. Для каждого $\tilde{z}^{(p)}$ выполняется преобразование, обеспечивающее подготовку фрагмента контейнера к встраиванию:

$$\tilde{z}_{-}^{(p)} = F_{double}[F_{int}(\tilde{z}^{(p)} + 0{,}5)] - 0{,}5,\ p = \overline{1, P}\ ,$$

где $F_{int}(\tilde{z}^{(p)} + 0{,}5)$ — функция, реализующая преобразование вещественных данных в диапазоне 0, ..., 1 к целочисленному формату представления с округлением в соответствии с используемой разрядностью данных.

На основе полученной реализации $\tilde{z}_{-}^{(p)}$, $p = \overline{1, P}$, определяется нормированный собственный вектор, соответствующий минимальному собственному числу

$$\varphi_n = r_{\min} / \sqrt{r_{\min}^{\mathrm{T}} r_{\min}}\ ,\ r_{\min} = \frac{1}{P} \sum_{p=1}^{P} (z^{(p)} - \tilde{z}^{(p)}).$$

Для его использования при последующем встраивании данных в фрагменты контейнера по-

лучим вектор φ_n^* на основе следующего преобразования:

$$\varphi_n^* = F_{double}[F_{int}(a_m\varphi_n + 0{,}5)] - 0{,}5,$$

где a_m — амплитуда вносимого искажения.

Наконец, при встраивании сообщения, образующего скрываемую последовательность данных ЦВЗ $d^{(p)}$, $p = \overline{1, P}$, формируется последовательность заполненных фрагментов контейнера $\bar{z}^{(p)}$, $p = \overline{1, P}$, представленных в целочисленном формате, на основе соотношения

$$\bar{z}^{(p)} = F_{int}(\tilde{z}_-^{(p)} + d^{(p)}\varphi_n^* + 0{,}5),\ p = \overline{1, P}. \qquad (2)$$

Таким образом, в данном случае процесс встраивания состоит в добавлении или вычитании (в зависимости от знака $d^{(p)}$, $p = \overline{1, P}$) малого по амплитуде сигнала, который соответствует округленному собственному вектору выборочной матрицы ковариации входной совокупности данных.

Получаемая на основе соотношения (2) последовательность используется для обучения ИНС (см. рис. 1, *б*), обеспечивающей восстановление ЦВЗ. Для этого выполняется преобразование, приводящее $\bar{z}^{(p)}$, $p = \overline{1, P}$, в вещественный формат данных

$$\bar{z}_d^{(p)} = F_{double}(\bar{z}^{(p)}) - 0{,}5,\ p = \overline{1, P}.$$

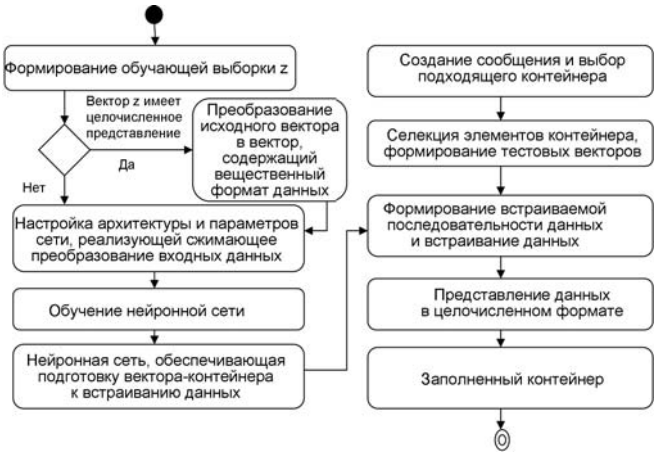


Рис. 3. Обобщенная схема встраивания информации при реализации предлагаемого нейросетевого метода

Рис. 4. Обобщенная схема извлечения информации при реализации предлагаемого нейросетевого метода

Обобщенную схему встраивания информации в рамках описанного подхода удобно представить в нотации языка UML, приведенной на рис. 3. Схема ССИ включает два основных этапа. На первом этапе формируются обучающие множества, настраиваются параметры ИНС и происходит процесс ее обучения. На втором этапе выбирается подходящий контейнер, формируются векторы входных воздействий и реализуется алгоритм встраивания.

Схема извлечения информации дана на рис. 4. Извлечение данных реализуется путем подачи элементов контейнера, содержащих встроенный ЦВЗ, на нейронную сеть, предварительно обученную для восстановления скрытой информации.

Описание разработанного программного комплекса

Программный комплекс (ПК) для создания цифровых водяных знаков разработан в среде Qt-4.6.0 и предназначен для работы в операционной системе Windows 7/Windows XP/Linux. В качестве языка разработки выбран C++. В ПК реализованы несколько альтернативных алгоритмов создания ЦВЗ и, прежде всего, алгоритмы, основанные на нейросетевых технологиях обработки информации, а также ряд специализированных упрощенных алгоритмов создания ЦВЗ для изображений, основанных на модификации уровней яркости относительно прогнозируемых значений. Функциональные возможности разработанного программного комплекса позволяют решать следующие задачи:

- осуществлять надежное встраивание/извлечение цифровых водяных знаков в виде черно-белых пиктограмм в ряд файлов мультимедийных форматов (bmp, png, jpg, tiff, avi и пр.);
- осуществлять детальную настройку предложенных алгоритмов встраивания ЦВЗ с тем, чтобы минимизировать результирующие искажения заполненных контейнеров;
- создавать пользовательские ЦВЗ (черно-белые изображения);
- обеспечивать удобство и простоту графического интерфейса, наглядность представления результатов работы алгоритмов встраивания и извлечения ЦВЗ.

Укрупненная структурная схема ПК представлена на рис. 5. На ней приведены наиболее важные модули, реализующие совокупность алгоритмов создания и встраивания ЦВЗ, а также совокупность алгоритмов для анализа контейнеров и извлечения встроенных меток.

В рамках основной части ПК на рис. 5 представлены:

- модули интерфейсной части приложения;
- модули реализации алгоритмов встраивания ЦВЗ;
- модули реализации алгоритмов декодирования файлов-контейнеров ЦВЗ;
- модули для создания и настройки параметров ЦВЗ;
- модули для конструирования и обучения нейронных сетей;
- вспомогательные модули математической обработки информации.

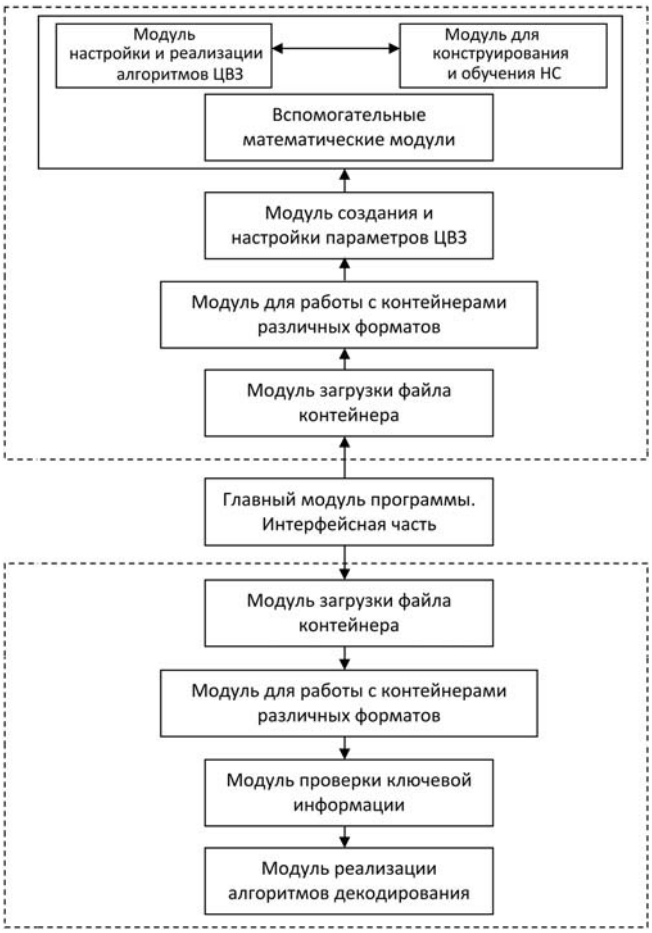


Рис. 5. Укрупненная структурная схема программного комплекса

При разработке программных модулей ПК проводилось разделение на классы описания интерфейсной части и классы описания и реализации логики работы алгоритмов встраивания/извлечении ЦВЗ. При этом для кодирования и декодирования контейнеров форматов использовались как разработанные классы, так и встроенные классы библиотеки Qt, позволяющие получить доступ к элементам контейнера и модифицировать их необходимым образом. Для работы с контейнерами видеоформатов использовалась библиотека ffmpeg [10], для работы с нейронными сетями — модули разработанной оригинальной библиотеки классов для моделирования и исследования нейронных сетей [11].

Процедура встраивания ЦВЗ состоит из следующих этапов. Сначала выбирается маркируемый файл. Осуществляется выбор или создание собственного ЦВЗ (графической пиктограммы). Определяется алгоритм, по которому будет проводиться встраивание метки, и при необходимости осуществляется корректировка его параметров. Для сохранения параметров, используемых при последующем восстановлении ЦВЗ, необходимо задать имя файла для хранения стеганографического ключа. После выполнения указанных процедур осуществляется встраивание ЦВЗ в указанный файл.

Процесс извлечения цифровой метки (или проверки наличия ЦВЗ) включает следующие шаги. Сначала указывается файл, предположительно содержащий ЦВЗ. Если его формат не поддерживается программой, выдается информационное сообщение с соответствующим предупреждением об ошибке. Если формат поддерживается, выбирается алгоритм извлечения и загружается файл, содержащий ключевую информацию. Внутренняя структура стеганографического ключа должна соответствовать предполагаемому алгоритму восстановления ЦВЗ. В соответствии с параметрами ключа корректируются настройки алгоритма восстановления. После выполнения указанных процедур осуществляется извлечение одного или нескольких ЦВЗ (точное число задается в стеганографическом ключе). Принятие решения относительно наличия или отсутствия ЦВЗ, а также корректность конечного результата оцениваются пользователем на основе восстановленного графического ЦВЗ.

Главное окно приложения (рис. 6, см. третью сторону обложки), содержит две вкладки "Встраивание ЦВЗ" и "Извлечение ЦВЗ". На вкладке "Встраивание ЦВЗ" в левой части расположены кнопки для перехода на страницы выбора и загрузки контейнера, создания ЦВЗ, выбора и настройки параметров алгоритмов, встраивания и визуализации результатов. В данном примере в качестве контейнера выступает изображение формата jpeg.

После загрузки защищаемого файла необходимо перейти на страницу создания ЦВЗ. В качестве цифровых меток в разработанном приложении используются черно-белые изображения — пиктограммы. В меню "Варианты создания ЦВЗ" предусмотрено несколько способов создания ЦВЗ, в том числе загрузка пиктограмм из файла, создание пользовательских пиктограмм или создание ЦВЗ из введенной пользователем строки символов.

На странице "Настройка алгоритмов" (рис. 7, см. третью сторону обложки) главного окна приложения осуществляется выбор и детальная настройка алгоритмов для встраивания ЦВЗ. Общим этапом для всех реализованных алгоритмов является определение числа экземпляров встраиваемых ЦВЗ, определение варианта выбора элементов контейнера для скрытия в них информации, задание имени стеганографического ключа. Возможное число встраиваемых ЦВЗ ограничивается доступным пространством ССИ, которое зависит от размера защищаемого контейнера, размера ЦВЗ и параметров выбранного алгоритма.

В программе предусмотрено два варианта выбора элементов контейнера: последовательно с начала файла или псевдослучайным образом равномерно по всей длине контейнера. Структура стеганографического ключа различается в зависимости от выбранного алгоритма встраивания ЦВЗ. Для ал-

горитма функционального встраивания ключевые данные включают:

- число встраиваемых экземпляров ЦВЗ;
- высоту и ширину встраиваемого ЦВЗ (пиктограммы);
- числовое значение варианта выбора фрагментов контейнера для встраивания (0 — последовательно, 1 — в соответствии с формированием псевдослучайных числовых последовательностей (ПСЧП)); начальные параметры ПСЧП, используемой при выборе фрагментов изображения для встраивания в них битов ЦВЗ;
- ширину и высоту кадра;
- вещественное значение амплитуды встраиваемого во фрагмент вектора;
- сведения об архитектуре НС, используемой для восстановления ЦВЗ (число входов, слоев, нейронов в каждом слое);
- параметры обученной ИНС (весовые коэффициенты и смещения).

После выбора алгоритма и настройки его параметров осуществляется переход на страницу "Встраивание" и запускается алгоритм встраивания цифровой метки.

Для восстановления ранее встроенного ЦВЗ необходимо перейти на вкладку "Извлечение ЦВЗ". На странице "Настройка алгоритмов" осуществляется загрузка файла с ключевой информацией. На основе данных стеганографического ключа автоматически выставляются настройки соответствующего алгоритма восстановления, создается НС необходимой конфигурации и происходит инициализация ее параметров. Процедура восстановления ЦВЗ запускается нажатием кнопки "Извлечь" на одноименной странице. По окончании работы процедуры формируется список пиктограмм восстановленных ЦВЗ. На рис. 8 представлено окно программы со списком из извлеченных меток, каждая из которых имеет различный уровень искажений. Цифровая метка считается корректно восстановленной, если характер ее изображения схож с исходной пиктограммой. В данном примере все метки можно считать корректно восстановленными.

Экспериментальные исследования

Тестирование программного продукта проводилось в части оценки качества контейнеров, получаемых в результате работы алгоритмов, а также надежности восстановления ЦВЗ. Качество алгоритма встраивания ЦВЗ в первую очередь обусловливается визуальной незаметностью искажающих изменений результирующего контейнера по сравнению с исходным. Надежность алгоритма восстановления ЦВЗ определяется его устойчивостью по отношению к различным трансформациям (случайным или преднамеренным), выполняемым над заполненном контейнером.

Пример исходного контейнера-изображения и ЦВЗ приведен на рис. 9. На рис. 10 (см. четвертую сторону обложки) представлены результаты работы нейросетевого алгоритма функционального встраивания в графические контейнеры форматов bmp, png, jpeg, tiff. Для каждого формата контейнера приводятся значения амплитуды вносимого искажения a_m (относительно диапазона значений входных данных ИНС 0, ..., 1), ошибки обучения НС, реализующей сжимающее преобразование для последующего встраивания E_{emb}, а также ошибки обучения НС, реализующей восстановление скрытой информации E_{xtr}. Как видно из рис. 10, в результате применении алгоритма были получены результирующие изображения очень высокого качества. При сравнении маркированных контейнеров

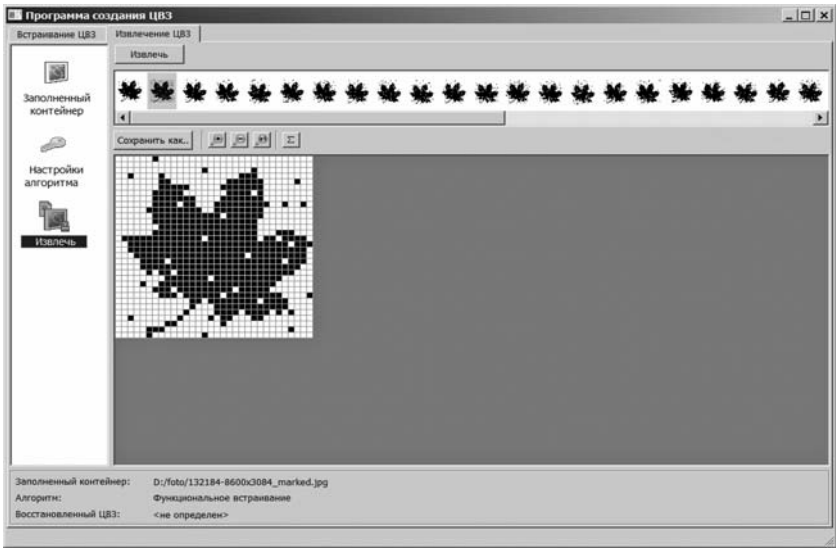


Рис. 8. Окно извлечения ЦВЗ и визуализации результата

а б

Рис. 9. Исходный контейнер "aircraft.jpg" (75.4 Кбайт) (*а*), размеры 900 × 588 пикселей, глубина цвета 24 бит/пиксель; изображение ЦВЗ "wm_0001.bmp" (190 байт), размеры 32 × 32 пикселей, 1 бит/пиксель (*б*)

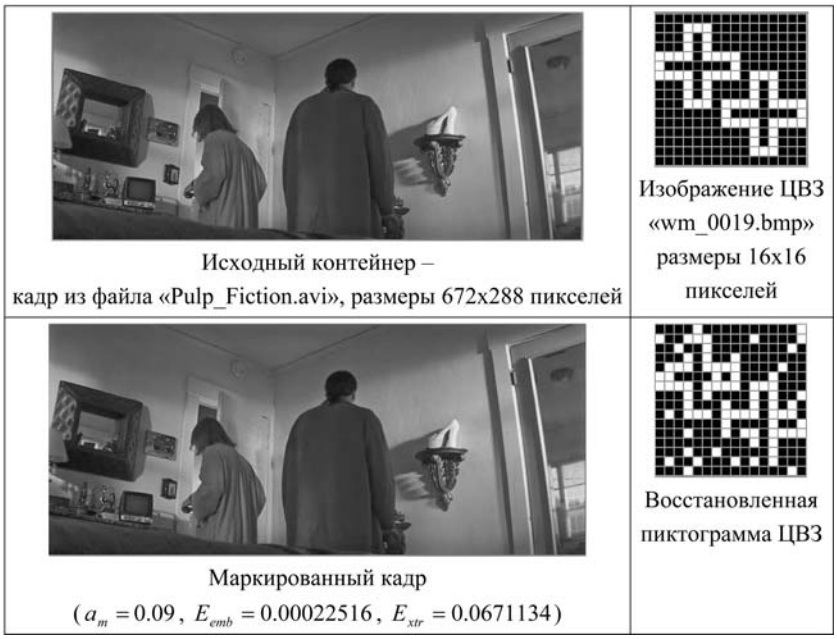


Рис. 11. Результаты работы алгоритма при создании ЦВЗ для видеофайла

с их оригиналами отыскать визуальные отличия практически невозможно. При этом исходная пиктограмма — ЦВЗ восстанавливается безошибочно для форматов контейнеров, не предусматривающих сжатие, и с определенными искажениями в случае JPEG-сжатия с потерями. Увеличение амплитуды вносимых искажений a_m позволяет повысить робастность встроенных меток, однако это не всегда желательно, поскольку может привести к потере качества защищаемого файла.

При встраивании ЦВЗ в видеофайлы выполняются следующие шаги. Контейнер разбивается на отдельные кадры, каждый из которых сохраняется на диске. Далее с помощью одного из выбранных алгоритмов осуществляется встраивание пользовательских данных в кадры, как в обычные изображения. По окончании встраивания полученные кадры снова объединяются в видеопоследовательность. Результаты встраивания ЦВЗ в кадр из файла формата avi приведены на рис. 11. Принципиальных отличий от полученных выше результатов встраивания в неподвижные изображения не наблюдается.

Как показали результаты экспериментов, разработанные алгоритмы (алгоритм функционального встраивания и нейросетевой алгоритм модификации уровней) продемонстрировали хорошие результаты по минимизации искажений при создании маркированных контейнеров, а также при восстановлении цифровых меток. При встраивании ЦВЗ формат контейнеров не искажался, а скрытые метки успешно восстанавливались даже после сохранения контейнеров в других форматах. Дополнительно для повышения устойчивости ЦВЗ при работе с графическими контейнерами целесообразно проводить разбиение изображения на блоки фиксированного размера и формировать входные/тестовые выборки для НС не из цветовых значений отдельных пикселей, а из усредненных цветовых значений блоков.

Разработанный ПК представляет собой удобный инструмент, поддерживающий работу с наиболее распространенными графическими и видеоформатами и позволяющий проводить гибкую настройку различных параметров алгоритмов для получения требуемого результата (минимизации визуальных искажений или повышения робастности ЦВЗ). Реализованные в рамках ПК алгоритмы обработки информации на основе использования ИНС могут быть легко адаптируемы для новых форматов контейнеров без необходимости принципиального изменения общей схемы создания ЦВЗ.

Работа выполнена при поддержке Фонда содействия развитию малых форм предприятий в научно-технической сфере (проект № 8501р/13581).

Список литературы

1. **Бахрушин А. П.** Спектральный анализ видеокадров на основе системы импульсных функций с целью синхронизации процессов внедрения и поиска цифровых водяных знаков // Вестник ТОГУ. 2008. № 4 (11).

2. **Мельников Ю. П., Теренин А. В., Погуляев В. Г.** Цифровые водяные знаки — новые методы защиты информации // Компьютерная неделя. 2007. № 48 (606). 25 декабря—31 декабря.

3. **Барсуков В. С., Шувалов А. В.** Еще раз о стенографии — самой современной из древнейших наук // Специальная техника. 2004. № 2.

4. **Kavithal V., Easwarakumar K. S.** Neural Based Steganography. Trends in Artificial Intelligence // PRICAI 2004. P. 429—435. URL: http://resources.metapress.com/pdf-preview.axd?code = q0bh4d8w9fjumdrj-&size = largest.

5. **Chuan-Yu Chang, Wen-Chih Shen.** Using counter-propagation neural network for digital audio watermarking. URL: http://dspace.lib.fcu.edu.tw/bitstream/2377/1060 /1/ce07ncs002006000074.pdf.

6. **Дрюченко М. А., Сирота А. А.** Нейросетевые модели и алгоритмы стеганографического скрытия информации // Тр. Российского научно-технического общества радиотехники, электроники и связи имени А. С. Попова. М., 2010. Т. 2. С. 335—338.

7. **Сирота А. А., Дрюченко М. А.** Нейросетевые модели и алгоритмы стеганографического скрытия информации // Информационные технологии. 2011. № 3. С. 41—49.

8. **Сирота А. А., Попов В. Г.** Свойства сходимости весов автоассоциативной двуслойной линейной нейронной сети при построении сжимающих отображений случайных векторов // Нейрокомпьютеры: разработка и применение. 2009. № 5. С. 3—11.

9. **Сирота А. А., Митрофанова Е. Ю.** Сходимость весов двухслойной линейной нейронной сети при построении оптимальных оценок случайных векторов // Нейрокомпьютеры: разработка и применение. 2011. № 7. С. 39—48.

10. **FFmpeg** libraries. URL: http://ffmpeg.org.

11. **Свидетельство** о государственной регистрации программы для ЭВМ "Программа для моделирования, обучения и тестирования нейронных сетей" № 2010613915 от 16.06.10, автор Дрюченко М. А.

УДК 004.942; 001.57; 519.876.5

Н. А. Емельянова, канд. физ.-мат. наук, доц.,
Казанский филиал Московского государственного университета путей сообщения (МИИТ),
Ф. М. Гафаров, канд. физ.-мат. наук, доц.,
Я. А. Сулейманов, аспирант,
Институт вычислительной математики и информационных технологий,
Казанский (Приволжский) федеральный университет,
Н. Р. Хуснутдинов, д-р физ.-мат. наук, проф.,
Институт физики, Казанский (Приволжский) федеральный университет,
e-mail: 7nail7@gmail.com

Математическая модель эволюции нейронной сети

Предложена математическая модель ветвления аксонов в процессе эволюции нейронной сети. Исследовано влияние активности нейронов на образование межнейронных связей, изучен процесс ветвления аксонов. Результаты исследования структурной пластичности в нейронных сетях могут быть применены широким кругом специалистов в области динамики сетей. Математическая модель и разработанная для ее описания компьютерная программа могут быть использованы для решения различных задач, связанных с ростом и динамическим развитием нейронных сетей.

Ключевые слова: *математическое моделирование, ветвление аксонов, конус роста, активность нейрона, нейронные сети*

Введение

В последние годы численные методы получили широкое распространение в исследованиях биологических систем. Идеализированная нейронная сеть, рассмотренная в данной работе, является простой моделью сети биологических нейронов корковой части головного мозга. Динамическое образование межнейронных связей в такой модели происходит за счет вещества, выделяемого самими нейронами с учетом ветвления аксонов. Образование нейронных сетей осуществляется путем соединения аксона с телом нейрона. Растущие аксоны и образованные от них ветви образуют сеть. В рамках используемой модели пренебрегается формой сомы нейрона, который рассматривается в виде окружности или сферы в зависимости от размерности задачи. Не рассматривается развитие дендритов, так как они являются частью сферы, которая рассматривается как сома нейрона. Положения нейронов фиксированы, все нейроны по своим формам, свойствам и поведению абсолютно одинаковы.

1. Нейробиологическая мотивация

Развитие связей между нейронами важно для правильного функционирования нервной системы. Известно, что аксоны могут устанавливать связи, распространяя ответвления от оси аксона к цели (нейрону). Во время развития аксона активность целевых нейронов управляет конусом роста, приводя к ветвлению аксона. Видеомикроскопия тонкого слоя ранней послеродовой коры показала, что конусы роста в различных областях коры имеют разное поведение [1]. Конус роста может продвигаться быстро и устойчиво, непрерывно изменяя форму. Ветви аксона со своими конусами роста развиваются от оси аксона и растут к вышележащей сенсомоторной области коры. Развитие идет в тех точках, где конусы роста делают паузу. Эти области приостановки конуса роста являются областями ветвления аксонов. Этот факт хорошо подтверждается исследованием роста таламических аксонов в органотипических культурах бокового коленчатого ядра (lateral geniculate nucleus LGN) и зрительной зоны коры головного мозга [1]. Появление несколькими часами спустя ветвления позади конуса роста означает, что полученные из цели сигналы могут вызвать приостановку конуса роста и инициировать рост промежуточных ветвей аксона. Конус роста прекращает свое движение, поскольку внешнее воздействие с разных сторон от целевых нейронов не дает точного сигнала, указывающего направление движения. Во время остановок конуса роста, которые могут длиться от 1 до 30 ч, аксон становиться в среднем в 6 раз больше того, который движется, и имеет место большое распространение ламеллиподии. Затем расширенный ламеллиподиум реорганизуется, формируя новый конус роста [1—3].

2. Математическая модель

В нашей модели используется термин AGM (*axon guidance molecules*) — совокупность всех типов молекул, выпущенных из целевых нейронов, которые участвуют в управлении аксона. Для описания распространения вещества AGM в модели используется стандартное уравнение диффузии в пространстве. Концентрация вещества AGM, $c_i = c_i(\mathbf{r}_j - \mathbf{r}_i, t)$, в точке $\mathbf{r}_j$ в момент времени t, выделившегося из i-го нейрона в точке $\mathbf{r}_i$, удовлетворяет уравнению диффузии

$$\frac{dc_i}{dt} - D^2 \Delta c_i - kc = J_i(\mathbf{r}_j, t), \qquad (1)$$

где D^2 — коэффициент диффузии; k — коэффициент деградации, описывающий постоянное уменьшение концентрации. Решение этого уравнения имеет следующий вид:

$$c_i(\mathbf{r}_j - \mathbf{r}_i, t) = \int G_d(\mathbf{r}_j - \mathbf{r}_i, t)\, C_i^0(\mathbf{r}_i) d\mathbf{r}_i + \int_0^t dt_k \int G_d(\mathbf{r}_j - \mathbf{r}_i, t - t_k) j_i(\mathbf{r}_i, t_k) d\mathbf{r}_i, \qquad (2)$$

где $C_i^0(\mathbf{r}_i)$ — начальное распределение концентрации; $G_d(\mathbf{r}, t)$ — функция Грина,

$$G_d(\mathbf{r}, t) = \frac{1}{(4\pi t D^2)^{d/2}} \exp\left(-kt - \frac{\mathbf{r}^2}{4tD^2}\right). \quad (3)$$

В начальный момент времени AGM отсутствует и процессом управляет источник

$$J_i(\mathbf{r}_i, t_k) = a\delta^{(d)}(\mathbf{r}_j - \mathbf{r}_i) j_i(t), \quad (4)$$

локализованный на i-м нейроне. Параметр a описывает количество вещества AGM, выделяющегося из нейрона за единицу времени. Функция $j_i(t)$ описывает активность i-го нейрона в момент времени t и $j_i(t) \leqslant 1$.

Таким образом, получаем следующее выражение для концентрации:

$$c_i(\mathbf{r}_j - \mathbf{r}_i, t) = a \int_0^t dt_k G_d(\mathbf{r}_j - \mathbf{r}_i, t - t_k) j_i(t_k). \quad (5)$$

В рамках нашей модели активность нейронов подчиняется следующему дифференциальному уравнению:

$$\tau \frac{dj_i(t)}{dt} = -j_i(t) + f(j_i^{\text{ext}}(t) + \sum_{k=1,\, k\neq i}^{N} \omega_{ik} j_k(t)). \quad (6)$$

Здесь $j_i^{\text{ext}}(t)$ — внешний источник, зависящий от времени t; $j_i(t)$ — активность i-го нейрона, а ω_{ik} — вес, который определяет тип связи, описывает влияние k-го нейрона на i-й нейрон и может принимать три значения: −1, 0 и 1. Когда $\omega_{ik} = 1$, то связь возбуждающая, если $\omega_{ik} = -1$, то связь подавляющая, при $\omega_{ik} = 0$ влияние отсутствует. Нейрон не может установить связь с самим собой, поэтому $\omega_{kk} = 0$. Тип связи зависит от активности, т. е., если активность $j_i(t)$ больше порогового значения $j_i^{\text{ext}}(t)$, то $\omega_{ik} = -1$, и, наоборот, если $j_i(t)$ меньше либо равно $j_i^{\text{ext}}(t)$, то $\omega_{ik} = 1$.

Для описания динамики роста аксона определим радиус-вектор положения конца аксона i-го нейрона в момент времени t, который подчиняется дифференциальному уравнению

$$\frac{d\mathbf{g}_i(t)}{dt} = \lambda F(j_i) \sum_{k=1}^{N} \nabla c_k(\mathbf{g}_i - \mathbf{r}_k, t), \quad (7)$$

где $F(j) = \theta(j^{th} - j)$ — ступенчатая функция, зависящая от активности; пороговый параметр j^{th} определяет движение аксона: аксон движется, если активность j нейрона больше порогового значения активности j^{th}. Параметр λ описывает чувствительность аксона.

В начальный момент времени t координаты конца аксона равны координатам собственного нейрона, и активность $j_i(t)$ равна нулю. Активность нейрона задается параметром $j_i^{\text{ext}}(t)$. Между нейронами нет связи, и все веса ω_{ik} между k-м нейроном и i-м нейроном равны нулю.

Для реализации ветвления необходимы условия, при которых аксон ветвится. В нашей модели это моделируется следующим образом. Во время роста и движения аксона определяются скорость его роста и концентрация AGM на его конце. Для запуска механизма ветвления аксона проверяем выполнение трех условий. Во-первых, скорость роста аксона не должна превышать пороговое значение скорости роста аксона v_g. Во-вторых, выполнение условия того, что концентрация AGM на конце нейрона находится в диапазоне от $c_{\min}$ до $c_{\max}$. В-третьих, длина L_b участка аксона, который ветвится, должна быть больше порогового значения L_{th}.

Процесс ветвления аксона разделяется на три этапа. На первом этапе проверяется концентрация AGM с помощью уравнения (5) вокруг конуса роста на расстоянии r_b. Как показывают видеонаблюдения реальных нейронов, конус роста "ощупывает" пространство вокруг себя [4]. Для реализации этого явления в нашей модели конец аксона окружается сферой, на которой вычисляется значение концентрации AGM. Точки на сфере расположены на некотором угловом расстоянии друг от друга по ширине и долготе:

$$\begin{aligned} x_i &= x_a + r_b \sin\theta\cos\varphi, \\ y_i &= y_a + r_b \sin\theta\cos\varphi, \\ z_i &= z_a + r_b \cos\theta, \end{aligned} \quad (8)$$

где x_i, y_i, z_i — координаты точки на сфере; x_a, y_a, z_a — координаты конца аксона; углы θ и φ изменяются с шагом в 5°.

После получения значений концентрации AGM вокруг конца аксона наступает второй этап, на котором из множества значений концентрации AGM выбираем точку с максимальным ее значением. Далее из этого же множества выбираем все точки, которые удовлетворяют следующему условию:

$$|c_i - c_{\max}| \leqslant \varepsilon_b, \quad (9)$$

где c_i — значение концентрации точки i на сфере, $c_{\max}$ — максимальное значение концентрации. В результате получаем ветвление. Число полученных ветвей получается бóльшим, чем должно быть. Разница значения концентрации между точкой с максимальным значением концентрации и точками вокруг нее незначительна. Такое же явление наблюдается в природе — конус роста расплывается через определенное время [1]. В точках с максимальным значением концентрации сходятся филоподии и образуются ветви.

На третьем этапе происходит соединение ветвей. Из полученных в результате второго этапа ветвей создаем несколько подмножеств ветвей. Критерием отбора в подмножество является условие того, что расстояние между i-й и k-й ветвями меньше либо равно r_{ik}:

$$\sqrt{(x_i - x_k)^2 + (y_i - y_k)^2 + (z_i - z_k)^2} \leqslant r_{ik},$$

Рис. 1. Блок-схема ветвления аксона

где x_i, y_i, z_i — координаты конца i-й ветви; x_k, y_k, z_k — координаты конца k-й ветви. В каждом из полученных подмножеств ветвей отбираем одну ветвь, у которой значение концентрации AGM на конце ветви максимально по сравнению с другими ветвями в данном подмножестве. В результате получаем ветви аксона. Схематично алгоритм ветвления показан на рис. 1.

3. Результаты численного анализа

Для численного анализа модели, описанной выше, разработана компьютерная программа на языке C++ с визуализацией этих процессов на базе пакета OpenGL. Загрузка параметров нейронной сети происходит из файла. Специально разработанное приложение позволяет создавать и изменять параметры нейронной сети, такие как число нейронов, их расположение, начальную активность, шаг интегрирования и т. д. Карта изменения активностей нейронов в процессе численного моделирования считывается из специального файла, в котором указаны время, номер нейрона и значение активности.

Моделирование проводилось с различным числом нейронов. В первом численном эксперименте число нейронов $N = 3$, во втором эксперименте — $N = 5$. Каждый нейрон в начальный момент времени имеет одну ветвь. Для моделирования были использованы следующие значения параметров:

1. Количество вещества, испускаемого нейроном, за единицу времени $a = 10^{-5}$ нМ/с.
2. Коэффициент диффузии $D^2 = 6 \cdot 10^{-7}$ см^2/с.
3. Коэффициент, описывающий чувствительность аксона $\lambda = 4 \cdot 10^{-5}$ см^2/нМ$\cdot$ с.
4. Коэффициент деградации $k = 10^{-3}$.
5. Время релаксации активности $\tau = 1$ с.
6. Пороговое значение активности $j^{th} = 0{,}51$.
7. Скорость роста аксона при ветвлении $v_g = 5 \cdot 10^{-7}$ см/с.
8. Минимальная длина ветви при ветвлении $L_b = 0{,}0225$ см.
9. Минимальная концентрация вещества при ветвлении $c_{\min} = 10^3$ а. е. м.
10. Максимальная концентрация вещества при ветвлении $c_{\max} = 10^4$ а. е. м.
11. Начальная длина ветви $r_b = 5 \cdot 10^{-3}$ см.
12. Радиус сома $r = 5 \cdot 10^{-3}$ см.

Моделирование этой системы, описывающей рост нейронной сети без ветвлений, было описано в работах одного из авторов [6, 7]. В данной работе моделируется ветвление аксона при различном числе и расположении нейронов, с различной активностью и временем активации нейронов. В первом численном эксперименте смоделировано ветвление, в котором образуются две ветви. Нейроны расположены в вершинах равнобедренного треугольника (рис. 2).

У нейрона номер 0 растет аксон, этот нейрон обладает активностью $j = 0{,}15$. Он является неактивным нейроном. Нейроны 1 и 2 имеют активность $j = 0{,}52$ и являются активными нейронами. Образование ветвей происходит в момент времени $t = 28400$ с и показано на рис. 2, *б*. Создается симметричное распределение концентрации вещества AGM относительно оси аксона, как показано на рис. 3. Как только аксон дорастает до точки, находящейся между нейронами 1 и 2, рост прекращается. На рис. 3 показано контурное изображение градиента концентрации AGM. На рис. 3, *а* показана система из четырех нейронов, расположенных в углах квадрата, где наблюдается максимальное значение концентрации AGM. К центру концентрация уменьшается. На рис. 3, *б* показана система из трех нейронов. У нижнего нейрона растет аксон, два других нейрона имеют одинаковую активность, т. е. концентрация вещества AGM в пространстве симметрична относительно нижнего нейрона.

Как только прекращается рост аксона, включается механизм ветвления. В численном эксперименте видно (см. рис. 2, *в*), что ветви направлены в сторону нейронов 1 и 2, т. е., число ветвей равно числу нейронов, которые воздействовали на конец аксона.

Во втором численном эксперименте система состоит из пяти нейронов (рис. 4). В данном эксперименте образуются четыре ветви, так как на конец аксона воздействуют четыре нейрона. Механизм

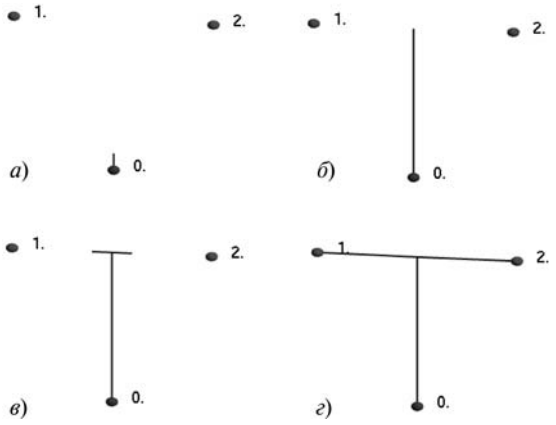
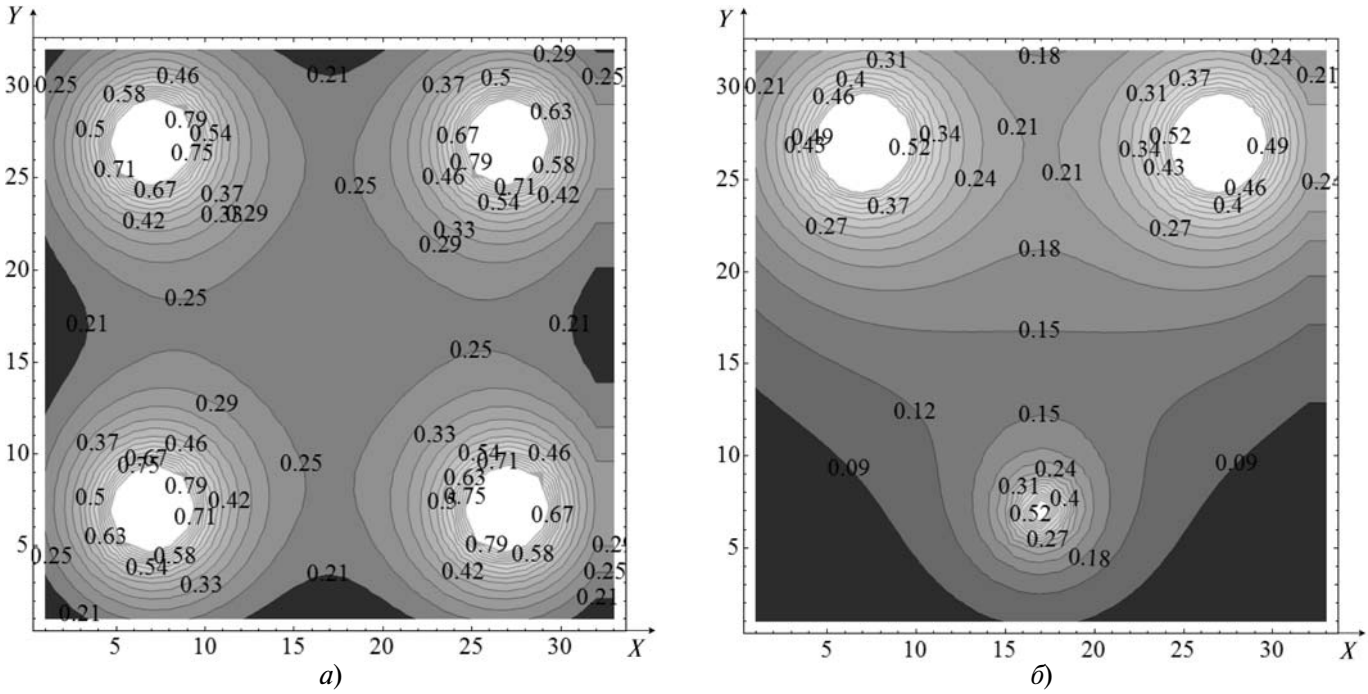


Рис. 2. Система из трех нейронов: нейрон 0 имеет активность $j = 0{,}15$, нейроны 1 и 2 имеют активность $j = 0{,}52$. Зафиксированы кадры: $t = 8000$ с (*а*), 28400 с (*б*), 31500 с (*в*), 37900 с (*г*)



а) *б*)

Рис. 3. Контурное изображение градиента концентрации AGM:
а — система из четырех нейронов; *б* — система из трех нейронов

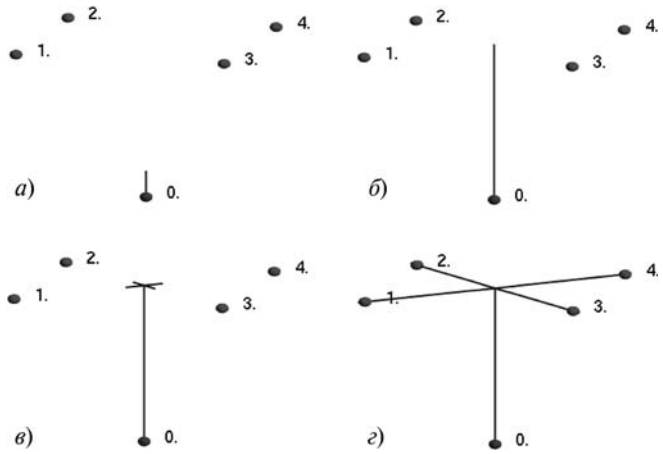


Рис. 4. Система из пяти нейронов: нейрон 0 имеет активность $j = 0{,}15$, нейроны 1, 2, 3, 4 — активность $j = 0{,}52$. Зафиксированы кадры: $t = 8200$ с (*а*), 27300 с (*б*), 27900 с (*в*), 62400 с (*г*)

ветвления в модели приводит к созданию ветвей, которые направлены в точки с максимальной концентрацией вещества AGM. Образование ветвей происходит в момент времени $t = 27300$ с, как показано на рис. 4, *б*. Этот пример иллюстрирует образование четырех ветвей.

Заключение

В работе предложена математическая модель роста нейронной сети, учитывающая особенности роста — ветвление аксонов в момент остановки роста. Модель базируется на уравнении теплопроводности, описывающем распространение вещества AGM. За основу модели процесса ветвления взяты экспериментальные наблюдения, описывающие поведение аксона во время своего роста. Предложенная математическая модель хорошо описывает процесс "ощупывания" пространства аксоном и выбор направления его роста в зависимости от концентрации окружающего вещества. В результате численного моделирования показано, что картина роста хорошо согласуется с данными экспериментальных наблюдений.

Используя предложенную модель, можно построить (вырастить) нейронную сеть с заранее заданными параметрами. Рост нейронной сети будет зависеть от временно́й структуры активности нейронов.

Список литературы

1. **Kalil K., Szebenyiand G., Dent E. W.** Common Mechanisms Underlying Growth Cone Guidance and Axon Branching // Journal of Neurobiology. 2000. Vol. 44, N 2. P. 145—158.
2. **Uesaka N., Hirai S., Maruyama T., Ruthazer E. S., Yamamoto N.** Activity Dependence of Cortical Axon Branch Formation: A Morphological and Electrophysiological Study Using Organotypic Slice Cultures // Journal of Neuroscience. 2005. Vol. 25, N 1. P. 1—9.
3. **Gibson D. A., Ma L.** Developmental regulation of axon branching in the vertebrate nervous system // Development. 2011. Vol. 138, N 2. P. 183—195.
4. **Dent E. W., Tang F., Kalil K.** Axon Guidance by Growth Cones and Branches: Common Cytoskeletal and Signaling Mechanisms // The Neuroscientist. 2003. Vol. 9, N 5. P. 343—353.
5. **Vitriol E. A., Zheng J. Q.** Growth Cone Travel in Space and Time: the Cellular Ensemble of Cytoskeleton, Adhesion, and Membrane // Neuron. 2012. Vol. 73. P. 1068—1081.
6. **Gafarov F.** Self-wiring in neural nets of point-like cortical neurons fails to reproduce cytoarchitectural differences / J. Integr. Neurosci. 2006. Vol. 5, N 2. P. 159—169.
7. **Gafarov F., Khusnutdinov N., Galimyanov F.** Morphless neurons compromise the development of cortical connectivity / J. Integr. Neurosci. 2009. Vol. 8, N 1. P. 35—48.

CONTENTS

This paper deals with the problem of localization accuracy estimation for indoor positioning systems based on nanoLOC (IEEE 802.15.4a) wireless sensors networks. The paper describes different approaches for positioning accuracy estimation.

Keywords: wireless sensors networks, indoor positioning systems, location estimation, nanoLOC^{TM}, IEEE 802.15.4a

The article is devoted to performance scaling assessment of the two most known multi-objective genetic algorithms SPEA2, NSGA-II. The special multi-objective test problems and indicators (see part 1) for quantitative assessment of quality Pareto sets are used.

Keywords: multi-objective optimization, Pareto's principles, Pareto front, performance indicators, multi-objective genetic algorithms SPEA2, NSGA-II

In this work the new algebraic approach realized on a basis of a "sign" technique, for an estimate of the solution of interval linear system at which the initial system is replaced with point (noninterval) systems in Euclidean space is offered. The specialized an algebraic approach allowing to evaluate of an unknown vector's coordinates for the point systems, similar by results of "external" estimate of set of the solutions received by methods of interval algebra is constructed. Application of a "sign" technique combines high computing efficacy with high quality of estimate sets of solutions, causing an estimate of sensitivity of linear interval systems.

Keywords: "outer" problem, "sign" technique, interval linear system, an estimates of the solution sets, point systems

The method of adjustment of associative intelligent system on entrance signals taking into account current loading of the system, expanding possibilities on processing of the information is considered. The mathematical formulation and algorithm of the decision of a problem is resulted. Results of modelling are reflected.

Keywords: associative intelligent system, neural network, adjustment, signal

Algorithms and data processing is implemented on the basis of their software system designed for digital watermarking as a means of protecting copyright of digital content objects. The basis for the construction of algorithms for digital watermarking is functional neural network model of data conversion.

Keywords: digital watermarks, neural networks, digital content, steganography

The mathematical model of branching axons during the neural net evolution is suggested. The influence of neuron activity on the interneural communications is investigated and the process of branching axons is studied. The results may be useful for specialists in the net dynamics. The mathematical model and the computer program realization of it may be used for solution different problems about net growth and dynamical evolution of neural nets..

Keywords: mathematical model, branching axons, growth cone, neuron activity, neural networks

Адрес редакции:

107076, Москва, Стромынский пер., 4

Телефон редакции журнала **(499) 269-5510**

E-mail: it@novtex.ru

Дизайнер *Т.Н. Погорелова*. Технический редактор *Е. В. Конова*.

Корректор *Т.В. Пчелкина*.

Сдано в набор 05.07.2012. Подписано в печать 16.08.2012. Формат 60×88 1/8. Бумага офсетная.

Усл. печ. л. 8,86. Заказ IT912. Цена договорная.

Журнал зарегистрирован в Министерстве Российской Федерации по делам печати, телерадиовещания и средств массовых коммуникаций.

Свидетельство о регистрации ПИ № 77-15565 от 02 июня 2003 г.

Оригинал-макет ООО "Адвансед солюшнз". Отпечатано в ООО "Адвансед солюшнз".

105120, г. Москва, ул. Нижняя Сыромятническая, д. 5/7, стр. 2, офис 2.