

Московский государственный университет
имени М.В. Ломоносова
Факультет вычислительной математики и кибернетики

На правах рукописи



Бычков Иван Алексеевич

**ПЛАНИРОВАНИЕ ОБМЕНОВ В СЕТЯХ С ТОПОЛОГИЕЙ
КОЛЬЦА С АРБИТРАЖЕМ ДЛЯ СИСТЕМ
РЕАЛЬНОГО ВРЕМЕНИ**

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

АВТОРЕФЕРАТ

диссертации на соискание учёной степени
кандидата физико-математических наук

Москва

2013

Работа выполнена на кафедре автоматизации систем вычислительных комплексов факультета вычислительной математики и кибернетики Московского государственного университета имени М. В. Ломоносова.

Научный руководитель: кандидат технических наук,
ведущий научный сотрудник
Костенко Валерий Алексеевич.

Официальные оппоненты: доктор технических наук,
профессор Колесов Николай Викторович;

кандидат физико-математических наук,
доцент Фуругян Меран Габидуллаевич.

Ведущая организация: ОАО «НИИВК им. М. А. Карцева».

Защита состоится «20» сентября 2013 г. в 11:00 на заседании диссертационного совета Д 501.001.44 при Московском государственном университете имени М. В. Ломоносова по адресу: 119991, ГСП-1, Москва, Ленинские горы, МГУ, второй учебный корпус, факультет вычислительной математики и кибернетики, аудитория 685. Желающие присутствовать на заседании диссертационного совета должны сообщить об этом за два дня по тел. +7 (495) 939-30-10 (для оформления заявки на пропуск).

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

Автореферат разослан «1» августа 2013 г.

Учёный секретарь

диссертационного совета Д 501.001.44

К. Т. Н., В. Н. С.



Костенко В. А.

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность темы. Вычислительные системы реального времени (ВСРВ) используются в самых разных областях человеческой деятельности, начиная от сравнительно простых бытовых устройств и заканчивая сверхсложными системами управления военными и космическими объектами. Важным подклассом ВСРВ являются вычислительные системы жёсткого реального времени. На временные характеристики функционирования таких систем накладываются ограничения, нарушения которых являются недопустимыми и которые рассматриваются как сбой работы системы. Примером такой системы может служить система управления ядерным реактором, в которой нарушения заданного времени реагирования на событие с высокой вероятностью может привести к катастрофе.

Одним из способов повышения надёжности в системах жёсткого реального времени является применение статического планирования. При статическом планировании расписание выполнения работ составляется ещё до запуска системы, на этапе её проектирования. Преимуществом статического планирования является простота аппаратного и программного обеспечения системы и гарантия выполнения режима жёсткого реального времени при отсутствии сбоев в работе. Поэтому при построении вычислительных систем жёсткого реального времени возникает задача построения статических расписаний.

Сложные технические системы, как правило, состоят из нескольких подсистем, каждая из которых выполняет определённую функцию. В настоящей работе рассматриваются вычислительные системы жёсткого реального времени, состоящие из нескольких подсистем, объединённых одной общей средой передачи данных. В качестве среды передачи данных в данной работе рассматривается кольцо с арбитражем.

Кольцо с арбитражем – это среда передачи данных, в которой информация передаётся по однонаправленному замкнутому контуру. Каждая подсистема, использующая кольцо с арбитражем для обмена информацией с другими подсистемами, выступает по отношению к кольцу с арбитражем в роли абонента – потребителя услуги, предоставляемой средой передачи данных, которая заключается в возможности передачи информационных сообщений от одного абонента другому абоненту (или сразу нескольким абонентам). Кольцо с арбитражем является одной из трёх топологий, поддерживаемых семейством открытых промышленных стандартов Fibre Channel. Актуальность решения задачи построения расписаний обменов в кольце с арбитражем подтверждается широким применением стандартов Fibre Channel в бортовых ВСПВ.

Одним из способов повышения надёжности кольца с арбитражем является применение концентратора. В кольце с арбитражем с концентратором на физическом уровне данные передаются по топологии «звезда», в центре которой находится концентратор, к которому подключены абоненты. В этом случае порядок передачи данных между абонентами определяется конфигурационными настройками концентратора и порядком подключения кабелей к его гнездам, что делает возможным изменение порядка абонентов в кольце с арбитражем. В случае, если концентратор не используется, изменение порядка абонентов также остаётся возможным на этапе проектирования системы. Для этого необходимо другим образом организовать каналы связи между абонентами. Таким образом, по крайней мере на этапе проектирования системы порядок абонентов в кольце с арбитражем может быть изменён.

При исследовании алгоритма построения расписаний оказалось, что порядок абонентов в кольце с арбитражем влияет на число информационных обменов, которые можно разместить в расписание. Так, можно привести пример, когда для одного порядка абонентов расписание, содержащее все ин-

формационные обмены из множества исходных информационных обменов (в дальнейшем – полное расписание), существует, а для другого порядка абонентов – нет. Этот эффект начинает проявляться тем чаще, чем выше загрузка канала. Поэтому при высокой загрузке канала выбор порядка абонентов является одним из решающих факторов, влияющих на возможность построения полных расписаний обменов. Поскольку на практике в большинстве случаев необходимо получать именно полные расписания обменов, то задача выбора оптимального порядка абонентов является актуальной.

Цель работы. Целью данной работы является разработка алгоритмов и инструментальных средств построения расписаний обменов и выбора порядка абонентов в кольце с арбитражем. Для достижения указанной цели необходимо решить следующие задачи:

- Разработать математическую модель информационных обменов в кольце с арбитражем и сформулировать постановки задач построения расписаний обменов и выбора порядка абонентов в кольце с арбитражем.
- Разработать алгоритм построения расписаний обменов в кольце с арбитражем.
- Разработать алгоритм выбора порядка абонентов в кольце с арбитражем.
- Провести исследование разработанных алгоритмов на реальных и модельных данных.
- На основе разработанных алгоритмов разработать инструментальную систему, позволяющую выполнять построение расписаний и выбор порядка абонентов в кольце с арбитражем.

Методы исследования. Для достижения указанных целей в работе были использованы методы математического моделирования, математического программирования, теории расписаний, эволюционные стратегии поиска решений, методы математической статистики и статистической проверки гипотез.

Основные результаты работы. Результаты диссертационной работы заключаются в следующем:

1. Предложена математическая модель информационных обменов в кольце с арбитражем, которая позволяет описывать различные способы построения кольца с арбитражем. В рамках разработанной модели предложена постановка задач построения расписания обменов и выбора порядка абонентов в кольце с арбитражем. Доказана NP-трудность поставленных задач.
2. Разработан алгоритм построения расписаний обменов в кольце с арбитражем на основе метода ветвей и границ. Получены достаточные условия, при которых алгоритм является точным. Получена верхняя оценка вычислительной сложности алгоритма.
3. Разработан алгоритм выбора порядка абонентов в кольце с арбитражем, основанный на эволюционных стратегиях. Получена верхняя оценка вычислительной сложности алгоритма.
4. На основе предложенных алгоритмов разработана инструментальная система построения расписаний, которая позволяет поддерживать процесс проектирования вычислительных систем реального времени, использующих в качестве среды передачи данных кольцо с арбитражем стандарта Fibre Channel.

Научная новизна. В диссертации предложена новая модель информационных обменов, в которой учитывается специфика кольца с арбитражем.

Разработан новый алгоритм построения расписаний информационных обменов в кольце с арбитражем, основанный на методе ветвей и границ. При фиксированной глубине перебора разработанный алгоритм имеет полиномиальную вычислительную сложность.

Разработан новый алгоритм выбора порядка абонентов в кольце с арбитражем, который во многих случаях позволяет найти расписание, содержащее

большее число обменов, чем при построении расписания для фиксированного порядка абонентов. Оба разработанных алгоритма обладают необходимой для практического применения точностью и вычислительной сложностью.

Практическая ценность. Разработано программное средство, в котором реализованы предложенные алгоритмы построения расписаний обменов и выбора порядка абонентов в кольце с арбитражем. Данное программное средство может быть использовано при проектировании сложных вычислительных систем, в которых информационные обмены выполняются по статическому расписанию, а в качестве среды передачи данных используется кольцо с арбитражем стандарта Fibre Channel.

Апробация работы. Результаты, представленные в работе, докладывались на научном семинаре лаборатории вычислительных комплексов кафедры АСВК факультета ВМК МГУ имени М. В. Ломоносова под руководством профессора Р. Л. Смелянского; на семинаре кафедры АСВК под руководством заведующего кафедрой член-корр. РАН Л. Н. Королёва; а также на следующих конференциях:

- III Всероссийская научная конференция «Методы и средства обработки информации» (МСО-2009), Москва, 6-8 октября 2009;
- VI Московская международная конференция по исследованию операций (ORM-2010), Москва, 19-23 октября 2010;
- VI Международная конференция «Параллельные вычисления и задачи управления» (РАСО'2012), Москва, 24-26 октября 2012.

Публикации. По теме диссертации опубликовано 7 печатных работ, в том числе работа в журнале «Математическое моделирование», который входит в перечень рецензируемых научных журналов ВАК РФ. Список работ приведён в конце автореферата.

Структура и объём диссертации. Диссертация состоит из введения, шести глав, заключения, списка литературы и девяти приложений. Объём ра-

боты — 170 страница (включая приложения), объём приложений составляет 65 страниц. Список литературы содержит 81 наименование.

ОСНОВНОЕ СОДЕРЖАНИЕ РАБОТЫ

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

В **первой главе** описана организация работы кольца с арбитражем, изложена формальная модель информационных обменов в кольце с арбитражем и сформулирована математическая постановка задачи построения расписаний обменов в кольце с арбитражем.

В разделе 1.1 описано понятие кольца с арбитражем. Кольцо с арбитражем – это среда передачи данных, в которой данные между абонентами передаются по однонаправленному замкнутому контуру. Кольцо с арбитражем предоставляет своим **абонентам** услугу, которая состоит в осуществлении **информационных обменов** между абонентами. Абоненты подключаются к кольцу с арбитражем с помощью портов. Порт служит интерфейсом связи между абонентом и кольцом с арбитражем. В целях идентификации каждому порту кольца с арбитражем ставится в соответствие два натуральных числа: номер и адрес. Эти величины задаются перед началом работы кольца с арбитражем и в процессе его работы остаются неизменными. Каждый порт может быть однозначно идентифицирован как по его номеру, так и по его адресу. Разница между этими двумя способами идентификации заключается в том, что адрес порта не зависит от порядка следования портов в кольце с арбитражем.

В разделе 1.2 описана задача построения расписания информационных обменов в кольце с арбитражем. Эта задача заключается в том, что заданы параметры кольца с арбитражем (в том числе порядок абонентов) и множество E^S информационных обменов, подлежащих выполнению в кольце с

арбитражем. Каждый информационный обмен выполняется путём передачи одного или нескольких сообщений.

Для каждого информационного обмена e из множества E^S известны значения следующих его атрибутов:

- $v(e)$ – количество слов данных в информационном обмене;
- $src(e)$ – адрес порта-передатчика информационного обмена;
- $DST(e)$ – множество адресов портов-приёмников информационного обмена;
- $s(e)$ – директивное время начала выполнения информационного обмена – момент времени, ранее которого не должно начинаться выполнение информационного обмена;
- $f(e)$ – директивное время окончания выполнения информационного обмена – момент времени, позднее которого не должно заканчиваться выполнение информационного обмена.

Совокупность этих величин позволяет однозначно вычислить:

- для каждого информационного обмена e величину $t(e)$ – длительность информационного обмена;
- для каждой упорядоченной пары информационных обменов (e_a, e_b) величину $\Delta_{min}(e_a, e_b)$ – *минимально допустимую задержку между информационными обменами e_a и e_b* , т. е. минимально допустимый интервал времени между началом выполнения информационного обмена e_a и началом выполнения следующего за ним информационного обмена e_b .

В задаче требуется построить расписание информационных обменов, содержащее максимальное число обменов, т. е. нужно выбрать максимальное подмножество множества E^S исходных информационных обменов и для каждого информационного обмена в выбранном подмножестве указать момент

начала его выполнения. При этом к расписанию обменов предъявляются следующие требования:

1. Выполнение каждого информационного обмена должно начинаться не ранее директивного времени начала его выполнения и заканчиваться не позднее директивного времени окончания его выполнения.
2. Для каждой пары последовательно размещённых в расписании информационных обменов должны соблюдаться минимально допустимые задержки, т. е. интервал времени между началом выполнения одного информационного обмена и началом выполнения следующего за ним информационного обмена должен быть не меньше заданной величины.

В разделе 1.3 описаны два способа организации информационных обменов в кольце с арбитражем – *децентрализованный* способ и *централизованный* способ. При децентрализованном способе организации информационных обменов каждый информационный обмен представляет собой передачу одного сообщения. Передача каждого сообщения инициируется непосредственно абонентом, который осуществляет передачу данного сообщения. Для того, чтобы каждый абонент имел возможность в нужный момент инициировать передачу очередного сообщения, копии расписания обменов хранятся в памяти всех абонентов кольца с арбитражем. Поскольку кольцо с арбитражем является общим ресурсом, то корректная передача сообщений в нём возможна лишь при условии выполнения всеми портами определённых правил. Эти правила описаны в подразделе 1.3.1.

При централизованном способе организации информационных обменов каждый информационный обмен представляет собой совокупность нескольких последовательно передаваемых сообщений. При этом абоненты кольца с арбитражем сами не инициируют информационные обмены – за это отвечает специальный выделенный абонент кольца с арбитражем, называемый контроллером кольца с арбитражем. Расписание обменов хранится только в

памяти контроллера, который в соответствии с этим расписанием рассылает другим абонентам команды, а абоненты кольца с арбитражем лишь выполняют эти команды. При использовании централизованного способа организации информационных обменов каждый информационный обмен представляет собой совокупность нескольких последовательно передаваемых сообщений. В подразделе 1.3.2 показано, каким образом вычисляются величины $t(e)$ и $\Delta_{min}(e_a, e_b)$ по известным значениям атрибутов информационных обменов для случая централизованного способа организации информационных обменов.

В разделе 1.4 описаны формальная модель информационных обменов и формальная постановка задачи \mathbf{T}^{SCH} выбора порядка абонентов в кольце с арбитражем.

Определение 1. Множеством исходных информационных обменов называется непустое конечное множество E^S , на котором определены следующие четыре функции:

- $s(e): E^S \rightarrow \mathbb{R}^+$ – директивное время начала выполнения информационного обмена e ;
- $f(e): E^S \rightarrow \mathbb{R}^+$ – директивное время окончания выполнения информационного обмена e ;
- $t(e): E^S \rightarrow \mathbb{R}^+$ – длительность выполнения информационного обмена e ;
- $\Delta_{min}(e_a, e_b): E^S \times E^S \rightarrow \mathbb{R}^+$ – функция минимально допустимых задержек.

Определение 2. Расписанием обменов (соответствующим множеству исходных информационных обменов E^S) называется множество $E \subseteq E^S$, на котором определена функция $s^*(e): E \rightarrow \mathbb{R}^+$ – время начала выполнения информационного обмена e . Причём для множества E и для функции $s^*(e)$ должны выполняться следующие два условия:

1. $\forall e \in E \Rightarrow (s(e) \leq s^*(e)) \wedge (f^*(e) \leq f(e))$ – интервал выполнения каждого информационного обмена должен лежать внутри его директивного интервала;
2. $\forall e_a \in E, \forall e_b \in next(e_a) \Rightarrow s^*(e_a) + \Delta_{min}(e_a, e_b) \leq s^*(e_b)$ – должны выполняться ограничения функции минимально допустимых задержек.

Здесь использованы следующие обозначения:

- $f^*(e) = s^*(e) + t(e)$ – время окончания выполнения информационного обмена e ;
- $next(e)$ – множество информационных обменов, непосредственно следующих за информационным обменом e . Множество $next(e)$ может содержать несколько информационных обменов, если время начала выполнения $s^*(e)$ для них одинаково.

Формальная постановка задачи T^{SCH} построения расписания обменов в кольце с арбитражем. Задано множество исходных информационных обменов E^S с определёнными на нём функциями $t(e)$, $s(e)$, $f(e)$ и функция $\Delta_{min}(e_a, e_b)$, определённая на множестве $E^S \times E^S$. Требуется найти расписание $E \subseteq E^S$, такое что $|E| \rightarrow \max$.

Во **второй главе** рассмотрены две физические топологии кольца с арбитражем. Определены параметры кольца с арбитражем, влияющие на его функционирование. Сделана формальная постановка задачи выбора порядка абонентов в кольце с арбитражем.

В разделе 2.1 рассмотрены следующие две физические топологии кольца с арбитражем: топология с концентратором и топология без концентратора. В случае использования топологии с концентратором порядок абонентов кольца с арбитражем, т. е. очерёдность следования абонентов друг за другом определяется: а) порядком подключения каналов связи ко входам/выходам концентратора и б) настройками концентратора. В топологии без концентратора порядок абонентов в кольце с арбитражем однозначно определяется тем,

какие пары портов соединены каналами связи. В обоих случаях на этапе проектирования системы изменение порядка абонентов в кольце с арбитражем не представляет большой трудности.

В разделе 2.2 приведено описание задачи выбора порядка абонентов в кольце с арбитражем. Данная задача рассматривается для различных топологий и различных способов организации информационных обменов в кольце с арбитражем. Для каждой из топологий определён свой набор параметров топологии, а для каждого способа организации информационных обменов – свой набор параметров способа организации информационных обменов. Итоговая совокупность параметров кольца с арбитражем образуется объединением этих параметров и зависит от того, какая используется топология и какой способ организации информационных обменов.

В зависимости от того, задано ли значение параметра топологии в задаче выбора порядка абонентов, или же, наоборот, оно должно быть найдено, каждый параметр кольца с арбитражем называется либо неизменяемым, либо изменяемым. В данной работе рассматривается лишь один изменяемый параметр – «порядок абонентов в кольце с арбитражем», а все остальные параметры считаются неизменяемыми. Если зафиксировать значения и неизменяемых, и изменяемых параметров, то кольцо с арбитражем будет полностью определено. Т. е. для такого кольца с арбитражем можно будет однозначно вычислить длительности выполнения информационных обменов и минимально допустимые задержки между ними.

Задача выбора порядка абонентов в кольце с арбитражем рассматривается в данной работе в следующей постановке. Заданы:

- фиксированный набор значений неизменяемых параметров кольца с арбитражем,
- совокупность информационных обменов, подлежащих выполнению в кольце с арбитражем.

Требуется выбрать такой порядок абонентов (который вместе с набором значений неизменяемых параметров кольца с арбитражем полностью определяет кольцо с арбитражем) и построить для данного кольца с арбитражем такое расписание, чтобы оно содержало максимально возможное число информационных обменов.

В разделе 2.3 приведена формальная постановка задачи выбора порядка абонентов в кольце с арбитражем. Задачу выбора порядка абонентов в кольце с арбитражем сокращённо будем обозначать символом \mathbf{T}^{ORD} . Исходные данные в задаче \mathbf{T}^{ORD} выбора порядка абонентов в кольце с арбитражем:

- $\bar{q} \in Q$ – фиксированный набор значений неизменяемых параметров;
- $R(\bar{q})$ – конечное множество допустимых значений порядков абонентов;
- E^S – непустое конечное множество исходных информационных обменов.

На множестве E^S определены функции:

- $s(e): E^S \rightarrow \mathbb{R}^+$ – директивное время начала выполнения информационного обмена e ;
- $f(e): E^S \rightarrow \mathbb{R}^+$ – директивное время окончания выполнения информационного обмена e ;
- $t(\bar{q}, r, e): Q \times R(\bar{q}) \times E^S \rightarrow \mathbb{R}^+$ – длительность выполнения информационного обмена e ;
- $\Delta_{\min}(\bar{q}, r, e_a, e_b): Q \times R(\bar{q}) \times E^S \times E^S \rightarrow \mathbb{R}^+$ – функция минимально допустимых задержек.

Определение 6. Индивидуальной задачей $\mathbf{T}^{\text{ORD}}(r)$, получающейся при фиксированном порядке абонентов r , будем называть задачу \mathbf{T}^{SCH} , для которой:

- $\widetilde{E}^S = E^S$. На множестве \widetilde{E}^S определены функции:
 - $\widetilde{s}(\tilde{e}) = s(e)$;
 - $\widetilde{f}(\tilde{e}) = f(e)$;
 - $\widetilde{t}(\tilde{e}) = t(\bar{q}, r, e)$;

$$- \tilde{\Delta}_{min}(\tilde{e}_a, \tilde{e}_b) = \Delta_{min}(\bar{q}, r, e_a, e_b).$$

Здесь все величины «с волной» относятся к задаче $\mathbf{T}^{\text{ORD}}(r)$, а «без волны» – к задаче \mathbf{T}^{ORD} . Элемент \tilde{e} множества \tilde{E}^S соответствует элементу e множества E^S . Значение величины \bar{q} задано из условий задачи \mathbf{T}^{ORD} .

Формальная постановка задачи \mathbf{T}^{ORD} .

$$\begin{cases} (r, E) = \arg \max(|E|) \\ r \in R(\bar{q}) \\ E \in \mathbf{T}^{\text{ORD}}(r) \end{cases}$$

Здесь запись $E \in \mathbf{T}^{\text{ORD}}(r)$ означает, что расписание информационных обменов E является оптимальным решением задачи $\mathbf{T}^{\text{ORD}}(r)$.

В **третьей главе** проведён обзор подходов к построению алгоритмов решения задач комбинаторной оптимизации и выполнен анализ их применимости к рассматриваемым задачам. Были рассмотрены следующие методы: метод полного перебора, жадные алгоритмы, метод ветвей и границ, генетические и эволюционные алгоритмы, муравьиные алгоритмы, метод динамического программирования, алгоритмы на основе нейросетей Хопфилда, ненаправленный случайный поиск, направленный случайный поиск.

На основании проведённого исследования для решения задачи построения расписания был выбран подход к построению алгоритма, основанный на применении ограниченного перебора с элементами метода ветвей и границ. Такой выбор позволяет избежать экспериментальной настройки большого числа параметров алгоритма, и в то же время позволяет регулировать вычислительную сложность алгоритма путём изменения ограничения на максимальную глубину перебора в дереве поиска.

Для решения задачи выбора порядка абонентов был выбран подход, основанный на эволюционных стратегиях. Такой выбор обусловлен тем, что данный подход позволяет строить алгоритмы, которые применимы к решаемым

мой задаче, предположительно должны демонстрировать высокую точность, а кроме того позволяют решать задачу выбора порядка абонентов с множеством допустимых порядков абонентов $R(\bar{q})$, функцией продолжительности выполнения информационных обменов $t(\bar{q}, r, e)$ и функцией минимально допустимых задержек $\Delta_{min}(\bar{q}, r, e_a, e_b)$, заданными в общем виде.

В **четвёртой главе** изложены предлагаемые алгоритмы решения рассматриваемых задач. Доказана корректность и завершимость алгоритмов. Приведены оценки их вычислительной сложности. Получены достаточные условия, при которых алгоритм построения расписания является точным.

Раздел 4.1 посвящён алгоритму решения задачи построения расписания обменов. В подразделе 4.1.1 описаны входные данные и результат работы алгоритма. Входными данными алгоритма \mathbf{A}^{SCH} являются:

- непустое множество исходных информационных обменов E^S с определёнными на нём функциями $s(e)$, $f(e)$, $t(e)$, $\Delta_{min}(e_a, e_b)$;
- $MaxRbk \in \overline{0, (|E^S| - 1)}$ – максимальная глубина перебора в дереве поиска.

Кроме того, должно быть задано отношение порядка \prec^{SEL} , определяющее порядок выбора информационных обменов при размещении их в расписание, и отношение порядка \prec^{DEL} , определяющее информационный обмен, подлежащий удалению в случае, если алгоритм просмотрел все допустимые варианты.

Результатом работы алгоритма является расписание $E = \mathbf{A}^{\text{SCH}}(E^S, MaxRbk)$ – приближённое решение задачи построения расписания обменов в кольце с арбитражем.

В подразделе 4.1.2 описан принцип работы предлагаемого алгоритма \mathbf{A}^{SCH} решения задачи построения расписаний обменов. Данный алгоритм основан на методе ветвей и границ со стратегией поиска в глубину. Процесс работы алгоритма \mathbf{A}^{SCH} можно представить как обход дерева поиска – ори-

ентированного графа (V, A) . Каждой вершине v данного графа соответствует некоторое расписание $sched(v)$ – набор размещённых в расписании работ (под *работой* понимается информационный обмен) – и набор $unsched(v)$ ещё не размещённых работ. Корнем дерева является вершина v_0 , соответствующая пустому расписанию: $sched(v_0) = \emptyset$, $unsched(v_0) = E^S$. Вершина v называется *тупиковой*, если хотя бы одна работа из множества $unsched(v)$ не может быть добавлена в конец расписания $sched(v)$, поскольку такое добавление повлечёт за собой нарушение директивного интервала данной работы или нарушение ограничения функции минимально допустимых задержек Δ_{min} . Из тупиковых вершин в дереве поиска рёбра не исходят. Вершина v , для которой $|unsched(v)| = 0$, называется *концевой*.

Если вершина v не является тупиковой, то из неё исходит $|unsched(v)|$ ориентированных рёбер, каждое из которых соединяет вершину v с вершиной w такой, что $sched(w) = sched(v) \cup e$, $unsched(w) = unsched(v) \setminus e$, $\forall e \in unsched(v)$. Таким образом, каждому ребру u в графе поиска соответствует некоторая работа $e_u \in E^S$, а переходу по ребру $u = (v, w)$ соответствует перемещение работы e_u из множества неразмещённых работ $unsched(v)$ в расписание $sched(v)$. При этом работе e_u назначается время начала выполнения $s^*(e_u)$, причём значение величины $s^*(e_u)$ не должно нарушать корректности расписания $sched(w)$. Существование такого момента времени гарантировано тем, что вершина v не является тупиковой, а значит любая работа из множества $unsched(v)$ может быть добавлена к расписанию $sched(v)$ без нарушения условий его корректности.

Алгоритм **A^{SCH}** осуществляет обход дерева в глубину в порядке, определяемом приоритетами рёбер, при необходимости возвращаясь по дереву поиска назад. Приоритет рёбер, исходящих из каждой вершины, определяется отношением порядка \prec^{SEL} . При обходе дерева алгоритм в первую очередь выбирает те рёбра, которые соответствуют меньшим значениям отношения

порядка \prec^{SEL} . Условием возврата к предыдущему уровню дерева является просмотр всех рёбер, исходящих из текущей вершины.

Для возможности контроля вычислительной сложности алгоритма введён параметр $MaxRbk$ – максимальная глубина перебора. Если алгоритм на какой-то итерации достиг вершины с глубиной n , то возврат по дереву поиска разрешается лишь до вершины с глубиной $(n - MaxRbk)$. При $MaxRbk = 0$ возвраты по дереву поиска запрещены, при $MaxRbk = (|E^S| - 1)$ алгоритм имеет возможность просмотреть всё дерево поиска.

Если, находясь в вершине v , алгоритм должен вернуться по дереву поиска на уровень вверх, однако это невозможно (вследствие того, что текущая вершина является корневой, или в силу ограничения на количество возвратов), то в этом случае алгоритм в соответствии с отношением порядка \prec^{DEL} выбирает некоторую работу e из множества $unsched(v)$ и удаляет её. При этом происходит изменение дерева поиска: рёбра и вершины, соответствующие работе e , перестают входить в дерево. Некоторые вершины могут перестать быть тупиковыми, если они были таковыми из-за того, что работа e не помещалась в соответствующие им расписания. Поскольку работа e исключается из рассмотрения, то за некоторыми вершинами, бывшими ранее тупиковыми, могут открыться новые ветви дерева.

После того как алгоритм удалил работу e из множества $unsched(v)$, он помечает все рёбра, исходящие из вершины v , как непросмотренные. То есть при дальнейшей работе алгоритма все рёбра, исходящие из вершины v , просматриваются алгоритмом заново (в соответствующем порядке), однако поиск происходит уже в изменённом дереве.

Алгоритм останавливается, когда найдена концевая вершина.

В подразделе 4.1.3 описаны функции и переменные, используемые в алгоритме. В подразделах 4.1.4 и 4.1.5 приведено пошаговое описание алгоритма и используемых в нём функций. В подразделах 4.1.6 и 4.1.7 доказаны кор-

ректность и завершимость алгоритма. В подразделе 4.1.8 доказаны условия, при которых алгоритм является точным.

Утверждение 4. Если $MaxRbk = (|E^S| - 1)$ и существует полное расписание $E \subseteq E^S$, то алгоритм построит расписание E' , которое будет полным.

Утверждение 5. Если для некоторого множества исходных работ E^S и значения параметра $MaxRbk$ алгоритм \mathbf{A}^{SCH} выдаёт полное расписание E , то при увеличении значения параметра $MaxRbk$ расписание E' , выдаваемое алгоритмом, также будет полным.

В подразделе 4.1.9 получена следующая верхняя оценка вычислительной сложности алгоритма: $O(|E^S|^{MaxRbk+2})$.

Раздел 4.2 посвящён алгоритму \mathbf{A}^{ORD} решения задачи выбора порядка абонентов в кольце с арбитражем. В подразделе 4.2.1 описаны входные данные и результат работы алгоритма. Входные данные для алгоритма \mathbf{A}^{ORD} совпадают с исходными данными для задачи \mathbf{T}^{ORD} выбора порядка абонентов в кольце с арбитражем.

В подразделе 4.2.2 приведено описание алгоритма \mathbf{A}^{ORD} . В алгоритме использована схема, близкая к классической схеме генетических алгоритмов. Каждая особь представляет собой порядок абонентов в кольце с арбитражем и кодируется перестановкой длины n , где n – число абонентов в кольце с арбитражем. Целевая функция определяется как максимальное число работ, которое можно разместить в расписание для порядка абонентов, соответствующего заданной особи. Значения целевой функции в алгоритме \mathbf{A}^{ORD} приближённо вычисляются алгоритмом \mathbf{A}^{SCH} . На вход алгоритма \mathbf{A}^{SCH} подаётся заданное множество исходных информационных обменов E^S , получающееся при фиксировании порядка абонентов r , для которого вычисляется целевая функция. В подразделе 4.2.2 описаны также параметры алгоритма, критерии останова, операция создания случайной особи, алгоритмы выполнения селекции, скрещивания и мутации особей и формирования новой популяции.

В подразделе 4.2.3 доказаны корректность и завершимость алгоритма. В подразделе 4.2.4 получена вычислительная сложность алгоритма.

В **пятой главе** изложены результаты численных исследований разработанных алгоритмов. В подразделе 5.1 приведено описание методики статистической обработке результатов экспериментов, основанной на методе проверки статистических гипотез.

В подразделе 5.2 описаны исходные данные, используемые в экспериментах. Исследования проводились на реальных и модельных исходных данных. Реальные исходные данные были получены на основании исходных данных, использующихся в существующих бортовых системах реального времени. Модельные данные генерировались случайным образом с последующей проверкой того, что они удовлетворяют определённым требованиям.

Использовались три вида реальных данных: неизменённые реальные данные, реальные данные с увеличенным объёмом передаваемой информации и шаблонные реальные данные.

Неизменёнными реальными данными называются множества исходных информационных обменов, используемые в существующих бортовых системах реального времени.

Реальными данные с увеличенным объёмом передаваемой информации называются неизменённые реальные данные, в которых объём передаваемых данных каждого информационного обмена увеличен в заданное число раз.

Шаблонными реальными данными (или просто шаблонными данными) называются исходные данные, полученные по неизменённым реальным данным с применением метода шаблонов. Этот метод позволяет получать исходные данных, имеющие частотно-фазовые характеристики, близкие к частотно-фазовым характеристикам исходных неизменённых реальных данных.

Использовалось два вида модельных данных: заведомо совместимые модельные данные и заведомо несовместимые модельные данные. Для заведомо

совместимых модельных данных существуют полные расписания, тогда как для заведомо несовместимых модельных данных полных расписаний не существует.

В подразделе 5.3 описаны результаты исследований. Подраздел 5.3.1 посвящён результатам исследования алгоритма построения расписаний \mathbf{A}^{SCH} . Для данного алгоритма были выбраны параметры и операции, которые использовались в исследованиях. Сначала было выбрано отношение порядка \prec^{SEL} , определяющее очередную работу, подлежащую добавлению в расписание. Исследование показало, что лучшим отношением порядка является отношение порядка \prec_{53}^{SEL} с функцией $g(e) = s(e) + s^*(e) + t(e)$. Затем было выбрано отношение порядка \prec^{DEL} , определяющее работу, подлежащую удалению из расписания. Было получено, что наилучшими являются отношения порядка \prec_{LSF}^{DEL} , \prec_{58}^{DEL} , \prec_{EDF}^{DEL} с функциями $g(e) = f(e) - t(e) - s^*(e)$, $g(e) = f(e) - t(e)$ и $g(e) = f(e)$ соответственно. Затем была выбрана максимальная глубина перебора $MaxRbk$, при которой время работы алгоритма не превосходит 5 мин. на одном ядре суперкомпьютера «Ломоносов», установленном в МГУ имени М. В. Ломоносова (процессор Intel Xeon 5570 Nehalem). Исследование проводилось на модельных данных с числом исходных информационных обменов, равным 25, 100, 500 и 2500, для которых были получены следующие значения максимальной глубины перебора: 5, 3, 1, 0 соответственно.

Была выполнена оценка точности алгоритма построения расписаний на выбранных значениях параметров. Исследование на трёх множествах реальных исходных данных показало, что для каждого из этих множеств можно увеличить объём передаваемых данных в 320, 830 и 2140 раз соответственно, и при этом алгоритм построит полное расписание. Исследование на заведомо совместимых модельных данных с числом исходных работ равным 100 показало, что даже на самом сложном классе исходных данных с вероятностью не менее 0,95 алгоритм в худшем случае не размещает только одну работу.

Была выполнена оценка числа выполненных возвратов по дереву поиска для различных типов исходных данных. Наибольшее значение возвратов составило 68 588 311 на заведомо совместимых модельных данных с числом исходных информационных обменов, равным 25, и максимальной глубиной перебора, равной 10.

Была выполнена оценка вклада составляющих алгоритма построения расписаний в его точность. Исследование показало, что основной вклад в точность алгоритма построения расписаний вносит использование отношения порядка \prec_{53}^{SEL} , а используемое отношение порядка \prec^{DEL} и механизм перебора влияют на результат в меньшей степени. При этом в ряде случаев применение механизма перебора позволяет сделать качественный шаг от несовместимого расписания к совместимому.

Подраздел 5.3.2 посвящён результатам исследования алгоритма **A^{ORD}** выбора порядка абонентов. В подразделе 5.3.2.1 выбраны исходные данные, на которых проводились исследования. Выбор осуществлялся на основании значения вероятности построения полного расписания для случайно выбранного порядка абонентов. Всего было выбрано семь различных наборов исходных данных. В подразделе 5.3.2.2 выбраны параметры, используемые в дальнейших исследованиях. В подразделе 5.3.2.3 выполнено исследование точности алгоритма. При стократном запуске алгоритма **A^{ORD}** на всех семи наборах исходных данных разработанный алгоритм нашёл порядок абонентов, для которого было построено полное расписание. В подразделе 5.3.2.4 выполнено исследование вычислительных затрат на выполнение алгоритма. Разработанный алгоритм сравнивался с алгоритмом случайного поиска. Исследование показало, что в большинстве исследованных случаев преимущество алгоритма **A^{ORD}** составляет 10–35%. В подразделе 5.3.2.5 показана стабильность алгоритма. В подразделе 5.3.2.6 исследовался вклад составляющих алгоритма в его точность. В качестве составляющих алгоритма рассматривались его

параметры. Исследование показало, что наибольшее влияние на точность алгоритма оказывает параметр «доля элитных особей от размера популяции при формировании новой популяции». Остальные параметры оказывают существенно меньшее влияние на результат работы алгоритма.

В **шестой главе** приведено описание разработанной инструментальной системы, в которой реализованы разработанные алгоритмы. В подразделе 6.1 сформулированы требования, которым должна удовлетворять инструментальная система. Система должна позволять пользователю:

- редактировать параметры кольца с арбитражем;
- редактировать множества информационных обменов в кольце с арбитражем (в т. ч. импортировать данные из файлов для шины МКИО ГОСТ Р 52070-2003);
- выполнять построение расписаний информационных обменов;
- выполнять выбор порядка абонентов в кольце с арбитражем.

Кроме того, инструментальная система должна:

- предоставлять пользователю возможность задания входных параметров алгоритмов построения расписаний обменов и выбора порядка абонентов в кольце с арбитражем;
- обеспечивать возможность работы как в консольном режиме, так и с использованием графического интерфейса пользователя;
- обладать кросс-платформенностью, т. е. обладать возможностью работать на вычислительных системах под управлением различных операционных систем.

В подразделе 6.2 приведена структура инструментальной системы. Инструментальная система состоит из трёх компонентов. Все компоненты системы написаны на языке C++. Суммарный объём кода составляет 10012 строк, 338 Кбайт.

Главным компонентом разработанной инструментальной системы является библиотека «система автоматизированного построения расписаний в кольце с арбитражем», реализующая предложенные в данной работе алгоритмы. Данная библиотека состоит из шести классов и одного пространства имён.

Второй компонент системы – два консольных приложения: **asch** и **aord**, которые позволяют из командной строки запускать соответственно алгоритмы **A^{SCH}** и **A^{ORD}**. Кроме того, эти программы используются в приложении с графическим интерфейсом пользователя.

Третий компонент системы – приложение с графическим интерфейсом пользователя, написанное с использованием кросс-платформенного средства разработки программного обеспечения *Qt* версии 4.7.

В **заклучении** приведены основные результаты диссертационной работы. В **приложении А** приведены определения используемых математических величин. В **приложении Б** описаны пространства поиска для рассматриваемых задач и доказана их конечность. В **приложении В** доказана *NP*-трудность в сильном смысле рассматриваемых задач. В **приложении Г** приведены детализированные правила передачи сообщений в кольце с арбитражем. В **приложении Д** приведена формализация основных понятий централизованного способа организации информационных обменов и описан механизм классов информационных обменов, упрощающий определение однотипных множеств информационных обменов. В **приложении Е** приведены параметры кольца с арбитражем для различных рассматриваемых в данной работе топологий и различных способов организации информационных обменов в кольце с арбитражем. В **приложении Ж** приведены формулы, позволяющие вычислить длительности передачи сообщений и минимально допустимые задержки между сообщениями для различных способов организации информационных обменов и различных топологий кольца

с арбитражем. В **приложении И** описан способ вычисления времени начала выполнения предварительного этапа и времени окончания выполнения завершающего этапа информационного обмена. В **приложении К** содержатся рисунки и таблицы, поясняющие результаты главы 5.

ПУБЛИКАЦИИ ПО ТЕМЕ ДИССЕРТАЦИИ

1. Бычков И. А., Костенко В. А. Возможность использования арбитражного кольца Fibre Channel в вычислительных системах реального времени // Программные системы и инструменты. Тематический сборник №8, М.: Изд-во факультета ВМиК МГУ, 2007. – С. 157-172.
2. Бычков И. А. Алгоритмы построения расписания обменов и выбора порядка устройств в кольце с арбитражем для систем реального времени // Сборник тезисов лучших дипломных работ 2008 года / Сост. Шевцова И. Г., Ильин А. В. – М.: Издательский отдел Факультета ВМиК МГУ им. М. В. Ломоносова; МАКС Пресс, 2008. – С. 81-82.
3. Бычков И. А., Костенко В. А. Задачи построения расписания обменов и выбора структуры сети для кольца с арбитражем // Методы и средства обработки информации: Третья Всероссийская научная конференция, Москва, 6-8 октября 2009 г.: Труды конференции / Под ред. Л. Н. Королёва. – М.: Издательский отдел ф-та ВМиК МГУ; МАКС Пресс, 2009. – С. 204-214.
4. Бычков И. А., Костенко В. А. Особенности задачи построения расписания обменов в кольце с арбитражем для систем реального времени // VI Московская международная конференция по исследованию операций (ORM2010): Москва, 19-23 октября 2010 г.: Труды / Отв. ред. П.С. Краснощёков, А. А. Васин. – М.: МАКС Пресс, 2010. – С. 285-287.

5. Бычков И. А. Представление математических моделей сред передачи данных жёсткого реального времени для задач статического планирования // Математическое моделирование. – 2011. – Т. 23. – №12. – С. 117-131.
6. Бычков И. А. Алгоритм выбора порядка абонентов в сети с топологией кольца с арбитражем // Параллельные вычисления и задачи управления (РАСО'2012). Шестая международная конференция, Москва, 24–26 окт. 2012 г. – Труды: в 3 т. – М.: ИПУ РАН, 2012. – Т. 1. – С. 238-249.
7. Бычков И. А. Алгоритм решения задачи построения расписаний для произвольно заданной функции минимально допустимых задержек // Мир компьютерной автоматизации: встраиваемые компьютерные системы, №2, 2013. – С. 52-64.