

На правах рукописи



Скородумов Юрий Михайлович

**НАЗНАЧЕНИЕ И ПЛАНИРОВАНИЕ ЗАДАНИЙ В РАСПРЕДЕЛЕННЫХ  
СИСТЕМАХ РЕАЛЬНОГО ВРЕМЕНИ**

Специальность: 05.13.11 – Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей

**АВТОРЕФЕРАТ**

диссертации на соискание ученой степени  
кандидата технических наук

Санкт-Петербург – 2016

Работа выполнена в государственном научном центре Российской Федерации АО «Концерн «ЦНИИ «Электроприбор», отдел вычислительной техники

- Научный руководитель:** доктор технических наук, профессор  
**Колесов Николай Викторович**
- Официальные оппоненты:** доктор технических наук, профессор,  
**Басыров Александр Геннадьевич**,  
Федеральное государственное казенное военное образовательное учреждение высшего профессионального образования «Военно-космическая академия имени А.Ф.Можайского» Министерства обороны Российской Федерации, начальник кафедры
- кандидат физико-математических наук,  
**Бахмуров Анатолий Геннадьевич**,  
Федеральное государственное бюджетное образовательное учреждение высшего образования «Московский государственный университет имени М.В.Ломоносова», старший научный сотрудник
- Ведущая организация:** Федеральное государственное автономное образовательное учреждение высшего образования «Санкт-Петербургский государственный университет аэрокосмического приборостроения», г. Санкт-Петербург

Защита состоится «28» сентября 2016 г. в 15:30 часов на заседании диссертационного совета Д 212.238.01 при Санкт-Петербургском государственном электротехническом университете "ЛЭТИ" им. В.И. Ульянова (Ленина) по адресу: 197376, Санкт-Петербург, ул. Профессора Попова, д.5.

С диссертацией можно ознакомиться в библиотеке Санкт-Петербургского государственного электротехнического университета "ЛЭТИ" им. В.И. Ульянова (Ленина) по адресу: 197101, Санкт-Петербург, ул. Проф. Попова, д.5 и на сайте [www.eltech.ru](http://www.eltech.ru)

Автореферат разослан «28» июня 2016 г.

Ученый секретарь  
диссертационного совета Д 212.238.01,  
к.т.н., доцент



Н.Л. Щеголева

## ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

**Актуальность темы диссертации.** Проблемы проектирования систем реального времени (СРВ) широко обсуждаются в современной научно-технической литературе как в России, так и за рубежом. Этот класс систем достаточно многообразен. К нему относятся системы обработки информации, управления подвижными объектами и технологическими линиями, навигации, связи, теле- и радиовещания. Примерами таких систем могут служить инерциальные и интегрированные навигационные системы, гравиметрические комплексы, системы освещения обстановки подводных аппаратов. Обычно СРВ принято характеризовать рядом определяющих особенностей. Среди них периодичность входного потока данных и, как следствие, периодичность всего вычислительного процесса. При этом период выбирается адекватным скорости протекания анализируемых и управляемых процессов. Кроме того, обычно отмечают необходимость привязки некоторых событий в процессах управления и обработки информации к заданным точкам или интервалам на временной оси посредством назначения директивных сроков или интервалов. Важно отметить, что современные СРВ отличаются высокой сложностью программного обеспечения, что заставляет в большинстве случаев с целью увеличения производительности обращаться к концепции распределенных вычислений. Можно утверждать, что процесс проектирования и эксплуатации СРВ обладает существенными особенностями и подчас ставит перед их разработчиками сложные вопросы, связанные, в частности, с организацией вычислений. Непременными составляющими этой проблемы являются назначение выполняемых заданий на процессоры и их планирование.

Проблемам назначения и планирования заданий в научно-технической литературе посвящено большое число публикаций. основополагающие результаты были получены в работах Liu C.L., Layland J.W., Coffman E.G., Cottet F., Stankovic J. A., Martello S., Toth P., Keller H., Топоркова В.В., Костенко В.А. и других авторов. Тем не менее, высокая размерность проблемы, характерная для рассматриваемой предметной области, необходимость проведения вычислений (а в ряде ситуаций и решения самой проблемы назначения и планирования) в реальном времени не позволяют применять многие известные подходы, нацеленные на получение оптимального результата. По этой причине широкое распространение на практике получили приближенные алгоритмы, позволяющие получить решение близкое к оптимальному за существенно меньшее время. Разработке и исследованию таких алгоритмов и посвящена настоящая диссертация, что позволяет считать ее тему актуальной.

**Объектом исследования** являются бортовые интегрированные системы обработки информации и управления в реальном времени.

**Предметом исследования** выступают методы назначения и планирования заданий в распределенных вычислительных системах обработки информации и управления в реальном времени.

**Цель работы** состоит в разработке и исследовании для распределенных вычислительных систем обработки информации и управления в реальном времени эффективных алгоритмов назначения и планирования заданий, близких по своим характеристикам к оптимальным при существенно меньшей вычислительной сложности.

Для достижения этой цели были поставлены и решены следующие **задачи**:

1. анализ современных методов назначения и планирования заданий;
2. разработка и исследование алгоритма назначения заданий на процессоры распределенных СРВ, в том числе и для избыточных систем;
3. разработка и исследование алгоритмов планирования заданий на основе концепции разрешимых классов систем;
4. разработка алгоритмов и программных средств для исследования эффективности и поддержки процедур назначения и планирования заданий распределенных СРВ;
5. подтверждение эффективности разработанных методов и программных средств по результатам их практической апробации.

**Методология и методы исследования.** Для решения поставленных задач использовались методы теории графов, дискретной математики, теории алгоритмов и теории надежности.

#### **Научная новизна**

1. Разработан основанный на графовом подходе приближенный алгоритм полиномиальной сложности для назначения заданий на процессоры распределенных СРВ, в том числе избыточных, отличающийся от известных приближенных алгоритмов учетом структуры заданий и позволивший повысить эффективность назначения в смысле принятого критерия в среднем на 10%.

2. Разработан метод планирования заданий для распределенных СРВ с одинаковой последовательностью посещений процессоров по критериям минимума общего времени выполнения плана и минимума максимального отклонения времен завершения заданий от их директивных сроков, представленный двумя версиями приближенного РКС-алгоритма. Алгоритмы отличаются от известных приближенных алгоритмов учетом характеристик планируемости системы и позво-

ляют при незначительном проигрыше по значению критериев существенно сократить время выполнения алгоритма.

3. Предложен комбинированный подход к назначению и планированию заданий для распределенных СРВ с разной последовательностью посещений процессоров, отличающийся от известных низкой вычислительной сложностью, а также использованием комбинации алгоритма сетевого упорядочивания и РКС-алгоритма.

#### **Практическая значимость и внедрение результатов**

1. Предложенные алгоритмы назначения и планирования заданий позволяют сокращать затраты на поиск эффективных вариантов реализации СРВ.

2. Разработанные программные средства позволяют автоматизировать процедуры назначения и планирования заданий СРВ.

3. Предложенные решения были применены при разработке в АО «Концерн «ЦНИИ «Электроприбор» систем навигации и освещения обстановки подводных аппаратов.

#### **Основные положения, выносимые на защиту:**

1. Алгоритм для назначения заданий на процессоры распределенных вычислительных СРВ, в том числе избыточных.

2. Метод планирования заданий для распределенных СРВ с одинаковой последовательностью посещений для двух критериев.

3. Комбинированный подход к назначению и планированию заданий для распределенной вычислительной СРВ с разной последовательностью посещений на основе алгоритма сетевого упорядочивания и РКС-алгоритма.

**Достоверность научных результатов и выводов** подтверждается использованием корректных математических приемов, сопоставлением аналитических результатов и данных, полученных в ходе математического моделирования и экспериментальных исследований, критическим обсуждением результатов работы на научно-технических конференциях.

**Апробация результатов работы.** Материалы диссертации докладывались и обсуждались на 6-й и 8-й Всероссийских мультikonференциях по проблемам управления (Дивноморское, 2013; 2015), на XIX Международном научно-техническом семинаре «Современные технологии в задачах управления, автоматизации и обработки информации» (Алушта, 2013), на XXVIII и XXIX конференциях памяти выдающегося конструктора гироскопических приборов Н.Н. Острякова (Санкт-Петербург, 2012; 2014), на XII Всероссийском совещании по проблемам управления (Москва, 2014), на Всероссийской конференции по проблемам управления в технических системах (Санкт-Петербург, 2015). Практическая апробация

результатов диссертационной работы осуществлена при разработке цифровых вычислителей в АО "Концерн "ЦНИИ "Электроприбор" в части алгоритмических и программных средств для поддержки процедур назначения и планирования заданий и исследования эффективности полученных решений.

**Публикации.** По материалам диссертации опубликовано 19 работ, из них 4 публикации в ведущих рецензируемых изданиях, рекомендуемых ВАК Минобрнауки РФ и 11 докладов в материалах всероссийских и международных конференций.

**Структура и объем диссертации.** Диссертация состоит из введения, четырех разделов, заключения и списка использованных источников, содержащего 97 наименований. Объем работы составляет 124 страницы, включая 35 рисунков и 16 таблиц.

## ОСНОВНОЕ СОДЕРЖАНИЕ РАБОТЫ

**Во введении** приводится общая характеристика работы – ее актуальность, научная новизна, практическая ценность, сведения об апробации работы и публикациях. Формулируются основные положения, выносимые на защиту.

**В первой главе** диссертации рассмотрены основные проблемы организации вычислительного процесса в системах реального времени, приводится анализ современных подходов при назначении и планировании заданий.

**Во второй главе** рассматривается проблема назначения заданий для распределенных систем реального времени и предлагается эвристический *основной* алгоритм назначения. Предполагается, что рассматриваемое множество задач представлено ориентированным ациклическим информационным графом  $G(S, T)$ , где  $S$  – множество ребер,  $T$  – множество вершин, причем вершины соответствуют задачам, а ребра отражают информационные связи, существующие между задачами. Каждой вершине и каждому ребру графа приписывается некоторый вес. Вес вершины отражает длительность решения соответствующей задачи, а вес ребра – длительность соответствующего информационного обмена.

В общем случае граф  $G(S, T)$  не является связным и состоит из  $m$  компонент связности  $g = \{g_i \mid i = \overline{1, m}\}$ , которые в дальнейшем будем называть заданиями, т.е. назначению на процессоры подлежат  $m$  независимых заданий  $\tau = \{\tau_i \mid i = \overline{1, m}\}$ . Каждое задание  $\tau_i$  состоит из задач  $\{\tau_{i,j} \mid j = \overline{1, n_i}\}$ , где  $n_i$  – число задач в  $i$ -м задании. Для всех задач известны длительности их исполнения на используемом типе процессора  $\{e_{i,j} \mid i = \overline{1, m}, j = \overline{1, n_i}\}$ . Все процессоры (процессорные модули с собственной памятью программ и данных) имеют одинаковую производительность, а

задачи и задания выполняются с одинаковым периодом  $T_{ex}$ . Причем за время  $T_{ex}$  любая задача может быть решена на одном процессоре, т.е.  $e_{i,j} \leq T_{ex}$ , однако все задачи из  $G(S,T)$  не могут быть решены на одном процессоре, что вынуждает применять распределенные вычисления. Число процессоров необходимых для решения всех заданий определяется в процессе работы алгоритма. Проблема формулируется как поиск среди возможных такого варианта разделения графа на подграфы, назначаемые на процессоры, который близок к оптимальному по критерию, представленному взвешенной суммой числа используемых процессоров  $|P|$  и числа каналов обмена  $|C|$ :

$$J = a_p |P| + b_c |C| \rightarrow \min, \quad (1)$$

где  $a_p$  и  $b_c$  – весовые коэффициенты. Распределение задач рассматривается в условиях детерминированного периодического потока входных данных, следствием которого являются ограничения реального времени. Первое связано с тем, что суммарное время  $E_i$  выполнения назначаемых на каждый  $i$ -й процессор задач не должно превышать периода поступления входных данных:

$$E_i = \sum_{k=1}^{m_i} e_{i,k} \leq T_{ex}, \text{ где } e_{i,k} \text{ – время выполнения } k\text{-й задачи } i\text{-го процессора. Согласно}$$

второму, суммарная длительность обменов  $D_l$  для каждого  $l$ -го канала обмена не должна превышать пропускной способности  $D_{max}$  канала обмена на периоде:

$$D_l = \sum_{k=1}^{m_l} d_{l,k} \leq D_{max}, \text{ где } d_{l,k} \text{ – длительность } k\text{-го обмена в } l\text{-м канале.}$$

Предлагаемый алгоритм состоит из трех частей: построение максимального остовного дерева (т.е. дерева, включающего все вершины графа и характеризующегося максимальным значением суммарного веса всех ребер), назначение задач на процессоры при обходе вершин графа и определение требуемого количества каналов обмена по результатам распределения задач по процессорам. При этом наиболее вероятным становится локализация поддеревьев остова в рамках отдельных процессоров и, как следствие, сокращение межпроцессорных обменов. В первой части алгоритма используется модификация известного алгоритма Прима. Во второй части алгоритма при обходе вершин используется представление остова исходного графа в виде бинарного дерева.

Исследование эффективности предложенного алгоритма осуществлялось путем сбора статистики с использованием программы случайной генерации примеров. Генерируемые наборы назначаемых заданий обладали свойствами, характерными для практических приложений, – древообразные графы заданий, а также графы с малым числом циклов. Результаты работы алгоритма сопоставлялись с

результатами наиболее эффективного эвристического кластерного алгоритма, при котором на один процессор назначаются в первую очередь пары задач, информационный обмен между которыми наиболее интенсивен. Результаты, полученные эвристическими алгоритмами, сопоставлялись с результатами оптимального алгоритма, который поставлял решение, оптимальное по критерию (1). В процессе исследования формировались статистики примеров, для которых алгоритм назначения давал результат, отличающийся от оптимального результата не более чем на 30% и соответственно 10% по выбранному критерию. На рисунке 1 приведены статистики, полученные для разных типов графов заданий. Выбор правильного соотношения между коэффициентами  $a_p$  и  $b_c$  определяется рассматриваемым приложением. Аргументом в построенных графиках является число базовых циклов, присутствующих в генерируемых графах заданий.

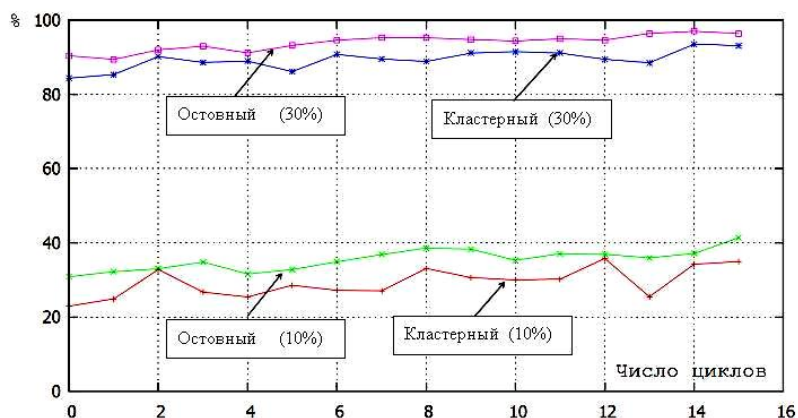


Рисунок 1. Эффективность алгоритмов назначения при  $b_c/a_p = 0,1$

Результаты исследования эффективности остовного алгоритма назначения показали, что в области его эффективного использования остовный алгоритм превосходит кластерный алгоритм и не более чем в 10% случаев проигрывает оптимальному алгоритму больше 30%. Для каждой точки графика случайным образом генерировалось по 500 тестовых примеров.

В диссертации предложено расширение алгоритма назначения для децентрализованной концепции отказоустойчивой распределенной СРВ. Концепция основана на принципе распределенного диагностического ядра, предполагающего взаимные проверки между процессорными модулями и распределенную процедуру принятия решений. Идея подхода состоит в том, что на каждый  $i$ -й процессорный модуль назначается избыточное число задач  $M_i$ . Среди них должны быть не только задачи  $m_i$ , которые исполняются при исправном состоянии системы, но и дополнительные задачи в количестве  $(M_i - m_i)$ , которые были назначены на другие



процессорные модули, но после отказа одного из них должны исполняться здесь. Таким образом, в номинальном режиме каждый модуль должен быть недогружен.

**В третьей главе** исследуется проблема планирования заданий для распределенных вычислительных систем из двух групп – соответствующих и несоответствующих модели flow shop. В первом случае для всех планируемых заданий характерна одинаковая последовательность посещений процессоров. На практике такая ситуация обычно возникает, когда в распределенной системе среди множества решаемых заданий можно выделить подмножества, каждое из которых использует информацию со своего набора датчиков. Постановка проблемы flow shop, рассмотренная в литературе, отличается от приводимой ниже и нацелена на конвейерные вычислительные структуры. В настоящей работе модель задания – направленный ациклический граф, содержащий в общем случае не один путь между любыми двумя вершинами и характеризующийся подмножеством входных вершин и одной выходной вершиной.

Планированию подлежат  $m$  независимых равноприоритетных заданий  $\tau = \{\tau_i \mid i = \overline{1, m}\}$ , обрабатывающих входные данные, поступающие с периодом  $T_{ex}$ . Каждое  $i$ -е задание состоит из  $n$  задач  $\tau_{i,j}$  длительностью  $e_{i,j}$   $j = \overline{1, n}$ . Все используемые процессоры (процессорные модули с собственной памятью программ и данных) из множества  $P$  имеют одинаковую производительность, а произведенное назначение заданий соответствует случаю flow shop. В данной работе это означает, что имеется  $m$  изоморфизмов  $\varphi_i : G_i(S_i, T_i) \rightarrow F(Q, P)$   $i = \overline{1, m}$ , где  $G_i(S_i, T_i)$  – граф межзадачных связей  $i$ -го задания,  $S_i$  – множество ребер,  $T_i$  – множество вершин (задач),  $F(Q, P)$  – граф межпроцессорных связей,  $Q$  – множество ребер,  $P$  – множество процессоров. Планирование осуществляется в условиях детерминированного периодического потока входных данных. Предполагается, что при назначении задач на процессоры было выполнено условие, гарантирующее независимость обработки разных порций входной информации (ограничение загрузки процессора на периоде поступления входных данных), что исключает образование очередей на процессоры из-за незавершенности обработки предыдущей порции входной информации.

Автором разработан РКС-алгоритм (алгоритм, основанный на разрешимых классах систем) планирования, применительно к двум критериям эффективности – минимуму общего времени выполнения плана и минимуму максимального отклонения от заданных директивных сроков. Алгоритм развивает известный подход к планированию в распределенных СРВ (Колесов Н.В., Толмачева М.В., Юхта П.В. Планирование вычислительного процесса в распределенных системах реаль-

ного времени с неопределенными длительностями решения задач // Теория и системы управления, № 4, 2012) по пути усложнения информационных графов заданий и использования расширенного множества специальных классов распределенных вычислительных систем с обновленными свойствами. Предложенные алгоритмы отличаются от известных учетом таких характеристик планируемости системы, как критический путь, отношение доминирования на множестве процессоров критического пути, а также близость к тому или иному специальному классу СРВ. Для определения классов предварительно введем на множестве процессоров отношение доминирования «>».

Определение. Процессор  $P_q$  доминирует над процессором  $P_r$  ( $P_q > P_r$ ), если  $\min_i e_{i,q} \geq \max_i e_{i,r}$ , ( $i = \overline{1, m}$ ), где  $e_{i,q}$  и  $e_{i,r}$  – длительности решения задач  $i$  и  $j$ , решаемых на процессорах  $q$  и  $r$  соответственно.

Общее свойство рассматриваемых далее специальных классов распределенных систем состоит в следующем: для любого задания, реализуемого в системе, критический путь одинаков, проходит по одним и тем же процессорам (обозначим его длину  $n^*$ ). Теперь приведем определения для каждого из четырех рассматриваемых специальных классов.

Класс 1. Множество процессоров критического пути представляет собой последовательность  $P_1 > P_2 > \dots > P_{n^*}$ , убывающую по отношению доминирования.

Класс 2. Множество процессоров критического пути представляет собой последовательность  $P_1 < P_2 < \dots < P_{n^*}$ , возрастающую по отношению доминирования.

Класс 3. Множество процессоров критического пути представляет собой пару соединенных последовательностей  $P_1 < P_2 < \dots < P_{h^*} > \dots > P_{n^*-1} > P_{n^*}$ , ( $1 \leq h^* \leq n^*$ ), первая из которых возрастает, а вторая убывает по отношению доминирования.

Класс 4. Множество процессоров критического пути представляют собой пару соединенных последовательностей  $P_1 > P_2 > \dots > P_{h^*} < \dots < P_{n^*-1} < P_{n^*}$ , ( $1 \leq h^* \leq n^*$ ), первая из которых убывает, а вторая возрастает по отношению доминирования.

Для каждого из классов существуют простые (линейной сложности) алгоритмы планирования по двум указанным выше критериям. Причем для первых трех классов эти алгоритмы являются оптимальными, поэтому в работе они названы разрешимыми, а для четвертого класса предложен приближенный алгоритм, опирающийся на верхнюю границу длительности плана для систем из этого класса.

В диссертации вводится обобщение использованного выше отношения доминирования:  $P_q \succ P_r \leftrightarrow f(q) \geq f(r)$ , где  $f$  – функция, аппроксимирующая последовательность медиан, вычисленных для интервалов длительностей задач, соответствующих процессорам критического пути. В диссертации в качестве  $f$  использована парабола ( $f(j) = aj^2 + bj + c$ ,  $j$  – номер процессора), как позволяющая наиболее удобно охарактеризовать систему. Далее, в зависимости от значений коэффициентов  $a$ ,  $b$  и  $c$  параболы, система соотносится с одним из четырех введенных выше специальных классов.

С использованием введенных понятий в диссертации формулируется эвристический алгоритм планирования – *PKC-алгоритм* для систем общего вида, не принадлежащих явно ни к одному из классов, состоящий в следующем.

*PKC-алгоритм планирования.*

1.  $C'$  присвоить  $C$ , где  $C$  – исходная система  $C = (P, \tau)$ , а  $C'$  – рассматриваемая на текущем шаге система.

2. Определение псевдокритического пути:

– определить для каждого процессора  $P_j$  системы  $C' = (P, \tau')$  (на первом шаге  $\tau' = \tau$ ) медиану  $\bar{e}_j$  на множестве длительностей решаемых на нем задач;

– найти в системе  $C' = (P, \tau')$  псевдокритический вычислительный путь  $p^*$ , который характеризуется наибольшим значением суммы  $\bar{e}(p^*) = \sum_{i=1}^{n^*} \bar{e}_j$  медиан для множеств времен решения задач.

3. На основе описанного ниже классификационного правила с использованием выделенного псевдокритического вычислительного пути определить, к какому классу наиболее близка рассматриваемая на данном шаге система  $C' = (P, \tau')$ .

4. Определить с использованием алгоритма планирования соответствующего класса для интервала свободных позиций плана одно из крайних заданий формируемого плана (первое для класса 1, последнее для класса 2), либо оба крайних задания (класс 3). Исключить из множества  $\tau'$  размещенные на данном шаге задания. Если множество  $\tau'$  не пустое, то перейти к п.2, иначе конец.

*Классификационное правило.* Методом наименьших квадратов определить коэффициенты аппроксимирующего полинома  $f(j) = aj^2 + bj + c$ , где  $j$  ( $j = \overline{1, n^*}$ ) – номер процессора псевдокритического пути длиной  $n^*$ . Если коэффициент « $a$ » больше нуля и абсцисса вершины параболы оказывается справа от интервала номеров процессоров, то система относится к классу 1, если слева – к классу 2, если внутри интервала – к классу 3. Если коэффициент « $a$ » меньше нуля и абсцисса вершины параболы оказывается справа от интервала номеров процессоров, то си-

стема относится к классу 2, если слева – к классу 1, если внутри интервала – к классу 4.

Результаты работы алгоритма сравнивались с оптимальным результатом, полученным методом полного перебора, или оценкой этого результата для примеров большой размерности. Кроме того, осуществлялось сопоставление с известным и наиболее эффективным эвристическим НЕН-алгоритмом, основанным на переборе ограниченного числа вариантов и не учитывающем свойств множества планируемых заданий. Оба приближенных алгоритма относятся к классу конструктивных эвристик. Для анализа эффективности была разработана программа случайной генерации примеров (наборов множеств планируемых заданий), при этом использовались два подхода.

В первом использовалась случайная генерация как графов заданий, так и длительностей составляющих их задач. При этом генерировалось по 500 примеров с числом заданий из интервала [3, 250], а длительность любой задачи формировалась как реализация случайной величины, равномерно распределенной в некотором интервале. В таблице 1 приведен показательный фрагмент, отражающий проигрыш исследуемых алгоритмов по отношению к оптимальному, для пяти структур, включающих по 10 заданий с 5, 10, 15, 25 и 50 задачами.

Таблица 1. Исследование эффективности РКС-алгоритма

Пример / Алгоритм	Проигрыш, %	
	НЕН	РКС
10 x 5	7,7	9,1
10 x 10	7,9	11,3
10 x 15	7,9	11,5
10 x 25	7,8	13,1
10 x 50	7,3	13,2
Средний	7,7	11,6

Видно, что в этом случае РКС-алгоритм проигрывает в среднем НЕН-алгоритму 3,9% (таблица 1). При этом вычислительная сложность РКС-алгоритма была существенно ниже, нежели у НЕН-алгоритма, что является одной из важнейших характеристик конструктивных эвристик. В частности при числе заданий больше 100, времена выполнения алгоритмов отличались более чем в 100 раз (рисунок 2).

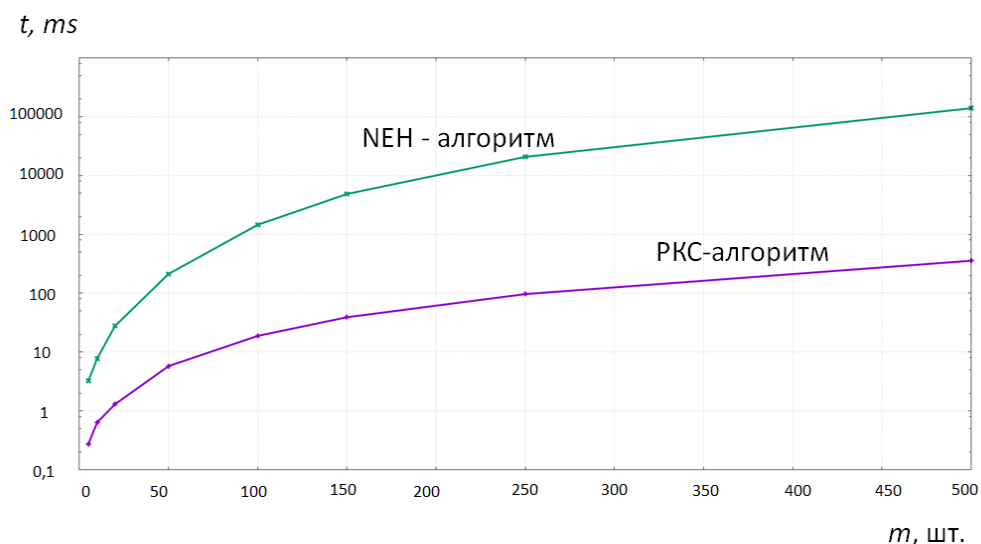


Рисунок 2. Времена выполнения NEH- и РКС-алгоритмов

Второй подход к анализу эффективности реализован на тестовых примерах Тейларда (Taillard), где предлагаемый алгоритм проигрывает оптимальному в среднем 15,7%, NEH-алгоритму – около 5,5%.

В этой главе также предлагается комбинированный подход к организации распределенных вычислений, когда проблемы назначения и планирования могут быть решены совместно методами сетевого и flow shop планирования. Такой подход к организации вычислений позволяет сокращать время выполнения вычислений в распределенных вычислительных системах, при этом не пренебрегая затратами на необходимые вычислительные ресурсы, что характерно для метода сетевого упорядочивания. Алгоритм можно представить как состоящий из двух этапов. На первом этапе осуществляется распределение вычислительных ресурсов для независимо выполняемых фрагментов заданий методами сетевого упорядочивания. Одним из результатов этого этапа будет формирование директивных сроков для второго этапа, когда при помощи алгоритмов flow shop планирования составляется расписание для совместно размещенных на процессорах фрагментов заданий.

**В четвертой главе** диссертации представлены результаты практической апробации предложенных методов назначения и планирования в распределенных СРВ. Приводятся примеры рассмотренных систем, основным среди которых является гидроакустический комплекс системы освещения обстановки (ГАС СОО) подводного аппарата. Кроме того, в диссертации содержатся сведения о применении методов в тренажере для операторов ГАС и в гидроакустической системе геологоразведки (ГАС ГР).

Особенностью информационного графа вычислительной системы ГАК СОО является наличие параллельных линейных участков, соответствующих заданиям, обрабатывающим данные от разных источников. Это позволяет применить модель flow shop при поярусном назначении задач. Таким образом, получаем распределенную вычислительную систему конвейерного типа.

Длительность полученного при помощи РКС-алгоритма плана, а также его проигрыш по отношению к оптимальному плану, полученному методом полного перебора, приведены в таблице 2. Также для сравнения в таблице приведена длительность наихудшего варианта упорядочивания заданий (верхняя граница), которая может быть достигнута, например, в случае использования произвольного расписания.

Таблица 2. Оценка эффективности планирования для ГАК

Алгоритм	РКС-алгоритм	Худший случай
Общее время выполнения плана Т, с	4,00	4,83
Проигрыш по отношению к оптим.	4%	26%
Проигрыш по отношению к пригл.	-	21%

Ниже приведены результаты, полученные для двух других систем при помощи инструментальной среды, позволяющей формировать конфигурацию событийной распределенной системы обработки информации с решением задачи назначения: тренажер оператора ГАК и гидроакустическая система геологоразведки (ГАС ГР). Информационный граф тренажера содержал 23 вершины. К нему были применены все предложенные и обозначенные в главе 2 алгоритмы назначения. Информационный граф ГАС ГР содержал 10 вершин. Результаты применения (полученные значения критерия (1), при  $b_c/a_p = 0,1$  для тренажера ГАК и  $b_c/a_p = 1$  для ГАС ГР) сведены в таблицы 3 и 4 соответственно.

Видно, что для обеих систем наилучший результат среди эвристических алгоритмов показал остовный алгоритм. Это объясняется характерным для данных приложений соотношением весовых коэффициентов  $b_c/a_p$ , отражающих сложность (стоимость) введения дополнительного процессора или канала обмена.

Таблица 3. Оценка эффективности назначения для тренажера ГАК

Название алгоритма	Оценка эффективности ( $J$ )	Проигрыш оптимальному
Кластерный алгоритм	95	72%
Остовный алгоритм	58	0,05%
Алгоритм полного перебора	55	—

Таблица 4. Оценка эффективности назначения в ГАС ГР

Название алгоритма	Оценка эффективности ( $J$ )	Проигрыш оптимальному
Кластерный алгоритм	14	133%
Остовный алгоритм	9	50%
Алгоритм полного перебора	6	–

## ОСНОВНЫЕ РЕЗУЛЬТАТЫ РАБОТЫ

В настоящей диссертационной работе рассмотрены теоретические и практические аспекты проблемы организации распределенных вычислительных систем реального времени. При этом получены следующие результаты.

1. Разработан основанный на графовом подходе приближенный алгоритм полиномиальной сложности для назначения заданий на процессоры распределенных СРВ, в том числе избыточных, отличающийся от известных приближенных алгоритмов учетом структуры заданий и позволивший повысить эффективность назначения в смысле принятого критерия в среднем на 10%.
2. Разработан метод планирования заданий для распределенных СРВ с одинаковой последовательностью посещений процессоров по критериям минимума общего времени выполнения плана и минимума максимального отклонения времен завершения заданий от их директивных сроков, представленный двумя версиями приближенного РКС-алгоритма. Алгоритмы отличаются от известных приближенных алгоритмов учетом характеристик планируемости системы и позволяют при незначительном проигрыше по значению критериев существенно сократить время выполнения алгоритма.
3. Предложен комбинированный подход к назначению и планированию заданий для распределенных СРВ с разной последовательностью посещений процессоров, отличающийся от известных низкой вычислительной сложностью, а также использованием комбинации алгоритма сетевого упорядочивания и РКС-алгоритма.
4. Разработаны программные средства для поддержки процедур назначения и планирования заданий в СРВ, а также для исследования их эффективности. Осуществлена апробация данных программных средств в различных разработках АО «Концерн «ЦНИИ «Электроприбор», в частности применительно к системам освещения обстановки подводных аппаратов.

## ПУБЛИКАЦИИ АВТОРА ПО ТЕМЕ ДИССЕРТАЦИИ

### Публикации в изданиях, рекомендуемых ВАК

1. Скородумов, Ю.М. Алгоритм независимого назначения иерархических заданий на процессоры в системе реального времени / Н.В. Колесов, Ю.М. Скородумов, М.В. Толмачева, П.В. Юхта // Вестник компьютерных и информационных технологий. – 2013. – № 6. – С.28 – 33.
2. Скородумов, Ю.М. Графовый подход к назначению заданий в распределенных системах реального времени / А.М. Грузликов, Н.В. Колесов, Ю.М. Скородумов, М.В. Толмачева // Известия Российской академии наук. Теория и системы управления. – 2014. – № 4. – С.28 – 38.
3. Скородумов, Ю.М. Нестационарные модели в задачах диагностирования вычислительных систем реального времени / А.М. Грузликов, Н.В. Колесов, Ю.М. Скородумов, М.В. Толмачева // Известия Российской академии наук. Теория и системы управления. – 2014. – № 6. – С.73 – 83.
4. Скородумов, Ю.М. Смешанное планирование заданий в распределенных системах реального времени / Н.В. Колесов, Ю.М. Скородумов, А.М. Грузликов, М.В. Толмачева // Вестник компьютерных и информационных технологий. – 2016. – № 5. – С.29 – 34.

### Другие публикации

5. Скородумов, Ю.М. Выполнение заданий в вычислительных системах реального времени // Материалы XV конференции молодых ученых «Навигация и управление движением». – 2013. – С.297 – 302.
6. Скородумов, Ю.М. Алгоритмы формирования вычислительного процесса в распределенных системах реального времени // Материалы XVII конференции молодых ученых «Навигация и управление движением». – 2015. – С.201 – 204.
7. Скородумов, Ю.М. Планирование вычислений в морских навигационных комплексах / Ю.М. Скородумов, П.В. Юхта // Материалы XXXI отраслевой научно-технической конференции молодых специалистов «Морское подводное оружие. Морские подводные роботы – вопросы проектирования, конструирования и технологий». – 2012. – С.71 – 76.
8. Скородумов, Ю.М. Выбор аппаратной платформы для реализации специфического вычислителя гидроакустического комплекса / Ю.М. Скородумов, П.В. Юхта, А.В. Шафранюк // Материалы XIV конференции молодых ученых «Навигация и управление движением». – 2012. – С.357 – 362.
9. Скородумов, Ю.М. Планирование вычислений в распределенных системах реального времени / Н.В. Колесов, Ю.М. Скородумов, М.В. Толмачева, П.В. Юхта // XXVIII Конференция памяти Н.Н. Острякова. – 2012. – С.57.
10. Скородумов, Ю.М. Назначение заданий на процессоры в системах реального времени / Н.В. Колесов, Ю.М. Скородумов, М.В. Толмачева, П.В. Юхта // XXVIII Конференция памяти Н.Н. Острякова. – 2012. – С.58.
11. Скородумов, Ю.М. Организация вычислений в распределенных системах реального времени / А.М. Грузликов, Ю.М. Скородумов, М.В. Толмачева // XIX Международный научно-технический семинар «Современные технологии в зада-



чах управления, автоматики и обработки информации», Алушта. – 2013. – С.210 – 211.

12. Skorodumov, Yu.M. Efficiency research of task allocation algorithms in distributed computing systems / A.M. Gruzlikov and Yu.M. Skorodumov // Proceedings of the international conference of young scientists «Automation and control», St. Petersburg, Russia, 21-22 november. – 2013. – p.46.

13. Скородумов, Ю.М. Назначение заданий при распределенных вычислениях / А.М. Грузликов, Н.В. Колесов, Ю.М. Скородумов, М.В. Толмачева // Материалы 7-й Всероссийской мультikonференции по проблемам управления, Дивноморское. – 2013. – Т.4. – С.34 – 37.

14. Скородумов, Ю.М. Нестационарные модели в задачах диагностирования вычислительных систем / А.М. Грузликов, Н.В. Колесов, Ю.М. Скородумов, М.В. Толмачева // XII Всероссийское совещание по проблемам управления, М.: Институт проблем управления им. В.А. Трапезникова РАН. – 2014. – С.7270 – 7281.

15. Скородумов, Ю.М. Планирование распределенных вычислений с минимизацией общего времени выполнения / А.М. Грузликов, Ю.М. Скородумов // Материалы XVI конференции молодых ученых «Навигация и управление движением». – 2014. – С.371 – 377.

16. Скородумов, Ю.М. Использование динамических моделей при мониторинге параллельных вычислений / А.М. Грузликов, Ю.М. Скородумов // Материалы XVI конференции молодых ученых «Навигация и управление движением». – 2014. – С.378 – 383.

17. Скородумов, Ю.М. Жадные алгоритмы планирования распределенных вычислений / Н.В. Колесов, Ю.М. Скородумов, М.В. Толмачева // XXIX Конференция памяти Н.Н.Острякова. – 2014. – С.360 – 366.

18. Скородумов, Ю.М. Комбинированный алгоритм планирования заданий в распределенных системах реального времени / Н.В. Колесов, Ю.М. Скородумов, М.В. Толмачева // Материалы 8-й Всероссийской мультikonференции по проблемам управления, Дивноморское. – 2015. – Т.4. – С.28 – 31.

19. Скородумов, Ю.М. Flow shop-планирование вычислений в распределенных системах реального времени / А.М. Грузликов, Н.В. Колесов, Ю.М. Скородумов, М.В. Толмачева // Материалы Всероссийской конференции по проблемам управления в технических системах, Санкт-Петербург. – 2015. – С.34 – 38.