На правах рукописи

Бобылева Ирина Владимировна

МЕТОД ПОПАРНОЙ ОБРАБОТКИ ЭЛЕМЕНТОВ ИНФОРМАЦИОННЫХ МАССИВОВ ДЛЯ МНОГОЗАДАЧНЫХ ВЫЧИСЛЕНИЙ В ГИБРИДНОМ ОБЛАКЕ

2.3.5. Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей

АВТОРЕФЕРАТ
диссертации на соискание учёной степени
кандидата технических наук

Самара — 2022

Работа выполнена в федеральном государственном автономном образовательном учреждении высшего образования «Самарский национальный исследовательский университет имени академика С.П. Королева» (Самарский университет) на кафедре информационных систем и технологий.

Научный руководитель:	**Востокин Сергей Владимирович** доктор технических наук, доцент, заведующий кафедрой программных систем, федеральное государственное автономное образовательное учреждение высшего образования «Самарский национальный исследовательский университет имени академика С.П. Королева», г. Самара.
Официальные оппоненты:	**Орлов Сергей Павлович** доктор технических наук, профессор, профессор кафедры вычислительной техники, федеральное государственное бюджетное образовательное учреждение высшего образования «Самарский государственный технический университет», г. Самара; **Трокоз Дмитрий Анатольевич** кандидат технических наук, доцент, проректор по научной работе, федеральное государственное бюджетное образовательное учреждение высшего образования «Пензенский государственный технологический университет», г. Пенза.
Ведущая организация:	**Международная межправительственная организация «Объединенный институт ядерных исследований»**, Московская обл., г. Дубна.

Защита состоится 25 мая 2022 года в 12:00 на заседании диссертационного совета 24.2.375.01 (Д 212.211.01) в ФГБОУ ВО «Рязанский государственный радиотехнический университет им. В.Ф. Уткина» по адресу:
390005, г. Рязань, ул. Гагарина, д.59/1.

С диссертацией можно ознакомиться в библиотеке ФГБОУ ВО «РГРТУ» и на сайте: http://rsreu.ru/post-graduate/dissertatsii/14027-item-14027.

Автореферат разослан «____» _____ 2022 г.

Ученый секретарь диссертационного совета 24.2.375.01 (Д 212.211.01) доктор технических наук, доцент

Пруцков Александр Викторович

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность. В основе информатизации лежат высокопроизводительные вычисления, связанные с обработкой больших массивов данных. Как правило, чем больше данных удается обработать в разумные сроки, тем большую ценность имеет извлекаемая из данных информация. Ограничивающим фактором повышения эффективности процесса обработки данных являются возможности вычислительной техники. Очевидный способ повышения эффективности – это увеличение компьютерного парка. Однако получить дополнительные вычислительные ресурсы можно арендуя виртуальные машины через сеть Интернет, а также используя простаивающие рабочие станции, кластеры, персональные компьютеры, подключенные к сети Интернет, в составе гибридного облака.

Вычисления в гибридном облаке невозможны без специальной организации вычислительного процесса, чтобы эффективно задействовать разнородные вычислительные ресурсы в совокупности. Их реализация требует решения научных задач в рамках исследования структуры алгоритмов и её отображения на вычислительную среду гибридного облака. Общая проблема отображения впервые была сформулирована и исследована в трудах Г.И. Марчука, В.В. Воеводина, Вл.В. Воеводина и др. применительно к задаче высокопроизводительных вычислений. В диссертации исследуется частная задача в рамках общей проблемы отображения, заключающаяся в поиске структур алгоритмов, которые легко отображаются на совокупность разнородных ресурсов гибридного облака (возможно за счет приемлемой потери эффективности при вычислении на типовых ресурсах), в тоже время выразительны при описании практических алгоритмов обработки данных.

Перспективный подход решения данной частной задачи отображения лежит в области вычислений с параллелизмом задач. Типовым применением параллелизма задач является поэлементная обработка информационных массивов в независимых задачах или так называемая чрезвычайная параллельность. Актуальное направление развития вычислений с параллелизмом задач – многозадачные вычисления, предложенные I. Raicu, I. Foster, Y. Zhao. В многозадачных вычислениях рассматриваются также информационные процессы, в которых имеются зависимости между задачами, а вычислительная сложность задач варьируется в широких пределах. Естественной моделью таких процессов и **объектом исследования** диссертации являются *параллельные алгоритмы попарной обработки элементов массивов данных*, когда параллелизм задач частично ограничен из-за невозможности одновременной обработки пар, включающих одинаковый элемент.

В качестве вычислительных ресурсов, для которых исследуется отображение алгоритмов попарной обработки, в диссертации рассматриваются разнородные ресурсы, получаемые с использованием технологий облачных

вычислений от нескольких поставщиков, составляющих гибридное облако. **Предметом диссертационного исследования** является разработка и исследование *метода отображения алгоритмов попарной обработки элементов массивов данных на архитектуру гибридного облака*.

Решение проблемы отображения включает несколько аспектов: системные средства организации вычислений (M. Livny, D.P. Anderson, И.В. Бычков, А.П. Афанасьев, О.В. Сухорослов и др.); алгоритмы планирования вычислительных процессов (В.В. Топорков и др.); языки программирования и моделирования вычислительных процессов (L. Lamport, A. Abbassi, А.В. Бухановский и др.); анализ информационных структур алгоритмов (В.В. Воеводин, Вл.В. Воеводин и др.). Диссертация посвящена исследованию информационных структур алгоритмов обработки данных. Это дает методическую основу использования технологий многозадачных вычислений для решения прикладных задач обработки данных и поэтому является актуальным.

Цель диссертационного исследования. Целью диссертационного исследования является развитие методов организации эффективных вычислений с использованием гибридных облачных систем в задачах попарной обработки элементов массивов данных.

Для достижения поставленной цели в диссертации решены следующие **задачи исследования**.

1. Анализ методов, моделей и программных средств, применяемых для глобально распределенной обработки данных.
2. Разработка метода синтеза алгоритмов попарной обработки элементов массивов данных с фиксированной структурой графа зависимостей задач.
3. Анализ алгоритмов попарной обработки элементов массивов данных и определение аналитических оценок ускорений для полученных алгоритмов.
4. Разработка алгоритмической модели многозадачного вычислительного процесса.
5. Применение алгоритмической модели многозадачного вычислительного процесса для реализации различных вариантов попарной обработки элементов массивов данных.
6. Экспериментальное исследование эффективности программ, построенных на основе алгоритмической модели многозадачного вычислительного процесса, при решении задачи попарной обработки элементов массивов данных в разнотипных вычислительных средах.
7. Разработка программной инфраструктуры для выполнения обработки элементов массивов данных с использованием гибридной облачной среды.

Научная новизна диссертационной работы. В диссертации получены следующие новые результаты.

1. В области разработки методов и алгоритмов для глобально распределенной обработки данных: метод синтеза параллельных алгоритмов для

попарной обработки элементов массивов данных, отличающийся возможностью синтеза алгоритмов с требуемыми свойствами при условии неизменяемости графа зависимостей задач в процессе вычислений; исследована вычислительная сложность трех алгоритмов типа «асинхронный круговой турнир» с использованием предложенного метода.

2. В области моделей создания программ и программных систем для параллельной и распределенной обработки данных: алгоритмическая модель многозадачного вычислительного процесса, отличающаяся новой интерпретацией известных моделей акторов и алгоритмических скелетов; впервые рассмотрены различные способы спецификации процессов типа «асинхронный круговой турнир» и их информационных структур с использованием предложенной модели.

3. В области разработки программной инфраструктуры для глобально распределенной обработки данных: предложена новая программная инфраструктура гибридного облака для многозадачной попарной обработки элементов массивов данных, позволяющая эффективно использовать собственные простаивающие и общедоступные внешние вычислительные ресурсы и легко адаптироваться для решения различных задач обработки данных.

Теоретическая и практическая значимость работы. Теоретическая значимость работы состоит в том, что на основе исследования взаимосвязей модели акторов Хьюитта, модели алгоритмических скелетов Коула и концепции многозадачных вычислений предложена новая модель многозадачного вычислительного процесса. Достоинством модели является спецификация многозадачного вычислительного процесса в виде последовательного алгоритма специальной структуры и возможность графического представления данной структуры. Это позволяет строить программные системы, эффективно реализующие вычисления и обработку данных на базе современных средств вычислительной техники. В качестве иллюстрации применения модели вычислительного процесса рассмотрен общий подход синтеза параллельных алгоритмов обработки данных типа «асинхронный круговой турнир».

Практическая значимость работы состоит в разработке программной инфраструктуры гибридного облака для многозадачной попарной обработки элементов массивов данных. Предложенная инфраструктура позволяет легко адаптировать программную систему для выполнения сортировки, частотного анализа, попарного сопоставления элементов неструктурированных данных. Практическим преимуществом программной инфраструктуры является возможность использования произвольных простаивающих вычислительных ресурсов (рабочих станций, персональных компьютеров, виртуальных машин гибридного облака) для снижения себестоимости обработки данных.

Прикладное значение также имеют результаты тестирования программных реализаций алгоритмов типа «асинхронный круговой турнир» при исполнении в различных вычислительных средах (многоядерном компьютере,

высокопроизводительной кластерной системе, кластере рабочих станций, кластере виртуальных машин гибридного облака), показывающие возможность эффективного применения предложенных методов создания программных систем многозадачных вычислений на практике.

Соответствие научной специальности. Работа соответствует следующим пунктам заявленной научной специальности: п. 8 «Модели и методы создания программ и программных систем для параллельной и распределенной обработки данных, языки и инструментальные средства параллельного программирования»; п. 9 «Модели, методы, алгоритмы, облачные технологии и программная инфраструктура организации глобально распределенной обработки данных».

Реализация результатов работы. Результаты диссертационного исследования получены и использованы при разработке программных систем и проведении вычислительных экспериментов на базе ресурсов облачной среды Самарского университета в проекте «Разработка фундаментальных основ аналитического синтеза регулярных и хаотических процессов в динамике космических аппаратов» (Госзадание № 9.1616.2017/4.6); для распределенной обработки данных в АО «РКЦ «Прогресс»; при проведении лабораторных практикумов по дисциплине «Архитектура современных распределенных систем» на кафедре информационных систем и технологий Самарского университета. Реализация результатов работы подтверждена актами о внедрении.

Методы исследований. В диссертационной работе используются элементы теории алгоритмов и теории графов, включая методы анализа сложности алгоритмов, методы параллельной обработки данных, формальные модели представления алгоритмов и информационных процессов на основе модели акторов Хьюитта и модели алгоритмических скелетов Коула.

Апробация работы. Основные результаты работы были представлены на следующих всероссийских и международных конференциях: V Всероссийской научно-технической конференции с международным участием «Актуальные проблемы ракетно-космической техники» («V Козловские чтения») (г. Самара, 2017); IV Международной конференции и молодежной школе «Информационные технологии и нанотехнологии» (ИТНТ-2018) (г. Самара, 2018); Международной научно-технической конференции «Перспективные информационные технологии» (ПИТ-2018) (г. Самара, 2018); XXI Всероссийском семинаре по управлению движением и навигации летательных аппаратов (г. Самара, 2018); 8-й Международной конференции «Распределенные вычисления и ГРИД технологии в науке и образовании» (Московская обл., г. Дубна, ОИЯИ, 2018); International Conference on Wireless Sensor Networks, Ubiquitous Computing and Applications (ICWSNUCA-2018) (India, Hyderabad, Gokaraju Rangaraju Institute of Engineering and Technology, 2018); V Международной конференции и молодежной школе «Информационные технологии и нанотехнологии» (ИТНТ-2019) (г. Самара, 2019); Международной

конференции «Суперкомпьютерные дни в России» (Russian Supercomputing Days 2019) (г. Москва, 2019); Международной молодёжной научной конференции «XV Королёвские чтения», посвящённой 100-летию со дня рождения Д.И. Козлова (г. Самара, 2019); Международной научно-технической конференции «Перспективные информационные технологии» (ПИТ-2019) (г. Самара, 2019); VI Международной конференции и молодежной школе «Информационные технологии и нанотехнологии» (ИТНТ-2020) (г. Самара, 2020); Международной научной конференции «Конвергентные когнитивно-информационные технологии» (г. Москва, 2020).

Авторский вклад. Все результаты, изложенные в диссертации и выносимые на защиту, получены автором лично. В работах, выполненных совместно, автору принадлежат части, относящиеся к моделированию информационных структур, синтезу и анализу алгоритмов обработки данных. Автор лично осуществлял проведение вычислительных экспериментов, обработку и интерпретацию полученных результатов.

Достоверность результатов работы. Достоверность полученных в работе оценок вычислительной сложности алгоритмов попарной обработки информационных массивов подтверждается корректными математическими выкладками и сопоставлением аналитических оценок вычислительной сложности с оценками, полученными в имитационных экспериментах. Эффективность алгоритмов, построенных с использованием предложенного метода и модели информационной структуры, подтверждена результатами нагрузочного тестирования их программных реализаций в различных вычислительных средах.

Основные положения, выносимые на защиту.

1. Метод синтеза параллельных алгоритмов для попарной обработки элементов массивов данных; результаты его применения для разработки алгоритмов типа «асинхронный круговой турнир» и анализа их ускорения.
2. Алгоритмическая модель многозадачного вычислительного процесса; результаты её применения для спецификации и программной реализации алгоритмов обработки данных типа «асинхронный круговой турнир».
3. Программная инфраструктура гибридного облака для многозадачной попарной обработки элементов массивов данных.

Публикации по теме диссертации. По теме диссертации опубликовано 17 работ, в том числе 2 статьи в журналах, рекомендованных ВАК РФ, 6 статей в рецензируемых изданиях, включённых в международную наукометрическую базу Scopus, одно свидетельство о регистрации программы для ЭВМ, 8 работ в материалах и трудах международных и всероссийских научных конференций.

Структура и объём работы. Диссертация состоит из введения, 4 глав, заключения и 3 приложений. Общий объем диссертации 160 страниц. Диссертация содержит 7 таблиц, 63 рисунка и список литературы из 138 источников.

СОДЕРЖАНИЕ РАБОТЫ

Во введении рассмотрены объект и предмет, сформулированы цель и задачи диссертационного исследования. Описаны полученные новые результаты, теоретическая и практическая значимость работы. Представлена реализация результатов работы и её апробация. Изложены основные положения, выносимые на защиту.

Первая глава посвящена анализу предметной области вычислений и обработки данных в глобальной сети Интернет. Рассмотрены парадигмы (грид-вычисления, облачные вычисления, многозадачные вычисления) и информационные модели (модель акторов, модель «портфель задач», алгоритмические скелеты), используемые при организации вычислений в сети Интернет. Даны примеры соответствующих приложений с указанием причин, по которым использование именно распределенной обработки в глобальной сети позволяет решить задачу эффективно.

Рассмотрены научные проблемы, возникающие при организации вычислений в сети Интернет. Отмечено, что разработка новых алгоритмов для вычислений в глобальной сети требует специальных методов спецификации и моделирования информационных структур алгоритмов, а перспективное направление развития указанных методов связано с адаптацией модели акторов и алгоритмических скелетов для описания многозадачных вычислений в гибридном облаке.

Вторая глава посвящена разработке метода синтеза многозадачных алгоритмов «асинхронных круговых турниров» для попарной обработки элементов информационных массивов, имеющих фиксированную структуру графов зависимостей задач. Вершины графов представляют задачи обработки одного элемента и задачи обработки пары элементов массива, а дуги графа представляют отношения непосредственного следования или предшествования задач информационного процесса.

Метод синтеза основан на приеме визуализации, в котором элементы данных (условно называемые командами-участниками турнира) объединены в цепочки, звенья которых перекладываются особым образом, а взаимное расположение звеньев друг над другом обозначает обработку пары элементов данных (игру команд-участников турнира). Процедура перекладки звеньев цепочек и получающийся в результате граф зависимостей задач для оптимизированного сортирующего турнира показаны на рисунках 1 и 2.

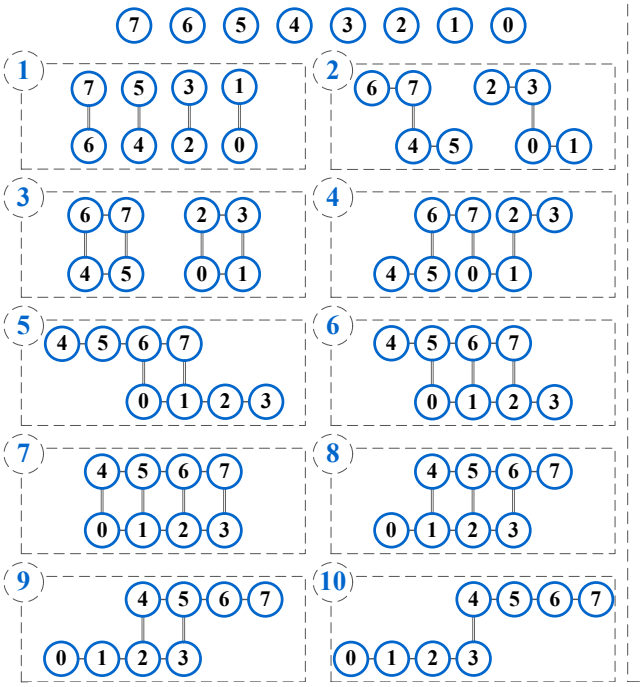

Рисунок 1 – Визуализация этапов перекладки звеньев при построении графа задач оптимизированного сортирующего турнира 8-ми команд

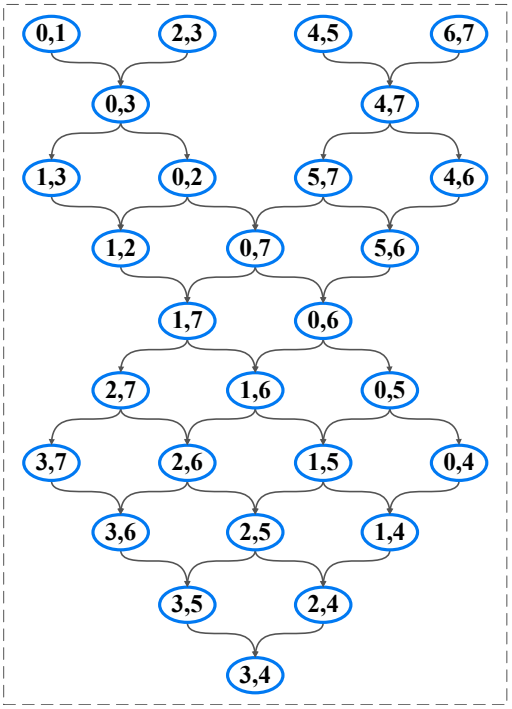

Рисунок 2 – Граф задач оптимизированного сортирующего турнира 8-ми команд, построенный согласно процедуре перекладки звеньев

Турнир, показанный на рисунках 1, 2, назван сортирующим, так как если под игрой турнира понимается обмен значениями двух элементов числового массива при нарушении их порядка, то по завершению всех игр турнира будет получен упорядоченный массив. Оптимизация заключается в уменьшении числа ярусов в графе рисунка 2 по сравнению с графом турнира, построенным по принципу пузырьковой сортировки.

Построение графа зависимостей задач в произвольных турнирах реализовано с помощью разработанного обобщенного алгоритма, параметром которого является процедура последовательного перечисления игр турнира. Для турнира на рисунках 1 и 2 рекурсивный алгоритм gen_optim для перечисления игр турнира game показан на рисунке 3. Также доказано утверждение о свойствах данного турнира.

Утверждение. Минимальное время в раундах $R(M)$ выполнения оптимизированного сортирующего турнира M команд $M = 2^k, k \geq 2, k \in \boldsymbol{N}$ равно $R(M) = \frac{3}{2}M - 2$. Ускорение при параллельной организации игр – $S(M) = \frac{M^2 - M}{3M - 4}$.

Доказательство. Пусть алгоритм вызывается для значений $M = 2^k$, $k \geq 2, k \in \boldsymbol{N}$. Будем отсчитывать уровни рекурсивного вызова процедуры gen_optim следующим образом. Самый глубокий уровень, на котором обрабатывается диапазон из 2-х чисел, – уровень 1; уровень, на котором обрабатывается диапазон из 4-х чисел, — уровень 2; уровень, на котором

обрабатывается диапазон из 8-и чисел, — уровень 3; и так далее до вызова процедуры gen_optim из алгоритма перечисления игр. Можно заметить, что начиная с уровня $i \geq 3$ некоторые игры, формируемые на уровне i, будут выполняться одновременно с играми, формируемыми нижележащим уровнем $i-1$. Определим сколько раундов игр будет сформировано именно на уровне i, без учета игр уровня $i-1$. Пусть m – длина цепочки номеров команд на некотором уровне рекурсии $i \geq 3$. Тогда от начала формирования новых раундов до выравнивания верхней и нижней подцепочек на уровне i будет сформировано $\frac{m}{4}+1$ новых раундов. После этого при дальнейшем движении подцепочек на уровне i будет сформировано $\frac{m}{2}-1$ раундов. То есть уровень рекурсии $i = log_2 m$ для диапазона длины m даст приращение количества раундов турнира $\frac{m}{2}+\frac{m}{4} = \frac{3}{4}m = \frac{3}{4}2^i$. Это означает, что общее число раундов $R(M) = 1 + \sum_{i=2}^{\log_2 M}\left(\frac{3}{4}2^i\right)$. Суммируя члены геометрической прогрессии, получаем $R(M) = \frac{3}{2}M - 2$. Тогда ускорение определяется как: $S(M) = \frac{M(M-1)}{2R(M)} = \frac{M^2-M}{3M-4}$.

```
рекурсивная процедура gen_optim
вход: range_begin, range_end

начало
    если range_end - range_begin = 0 то
    |   возврат в точку вызова gen_optim
    если range_end - range_begin = 1 то
    |   выполнить game(range_begin,range_end)
    |   возврат в точку вызова gen_optim
    range_begin_0 := range_begin
    range_end_0 := range_begin+(range_end-range_begin+1)/2-1
    range_begin_1 := range_end_0+1
    range_end_1 := range_end
    вызвать рекурсивно gen_optim(range_begin_0,range_end_0)
    вызвать рекурсивно gen_optim(range_begin_1,range_end_1)
    для i от range_end_1 до range_begin_1
    |   для j от range_begin_0 до range_end_0
    |   |   выполнять game(j,i)
конец
```

Рисунок 3 – Алгоритм gen_optim перечисления игр game в оптимизированном сортирующем турнире. Алгоритм вызывается в виде gen_optim(0, *M*-1) из обобщенного алгоритма синтеза графа зависимостей задач, *M* – количество команд

Метод синтеза многозадачных алгоритмов «асинхронных круговых турниров» для попарной обработки элементов информационных массивов также проиллюстрирован примерами синтеза и определения аналитических оценок ускорений для турнира без дополнительных ограничений и простого сортирующего турнира.

В третьей главе предложена алгоритмическая модель многозадачного вычислительного процесса. Модель базируется на придании алгоритму

специальной структуры, для которой отображение на заданный тип архитектуры компьютера выполняется известным способом. Этот подход напоминает представление алгоритмов в итеративно-параллельной, рекурсивно-параллельной или векторной форме. Однако в отличие от перечисленных форм алгоритмов, в качестве базовой формы для многозадачного представления алгоритмов взята модель акторов Хьюитта, для которой построена алгоритмическая интерпретация, а структура алгоритма задается при помощи алгоритмического скелета, аналогично подходу Коула (рисунки 4, 5, 6).

```
Algorithm( run() ) ->

AM<actor,message>(
   init(){set(message,actor,boolean)}
   recv(message,actor){
       send(message,actor)
       access(message)
   }
)
```

Рисунок 4 – Алгоритмический скелет акторных многозадачных алгоритмов (АМ)

```
начало
     выполнить init()
     для всех акторов a: a.active := false
     пока есть такие m, что m.active = true выполнять
     |    m.active := false
     |    m.act.active := true
     |    выполнить recv(m, m.act)
     |    m.act.active := false
конец
```

Рисунок 5 – Алгоритм run в определении алгоритмического скелета АМ

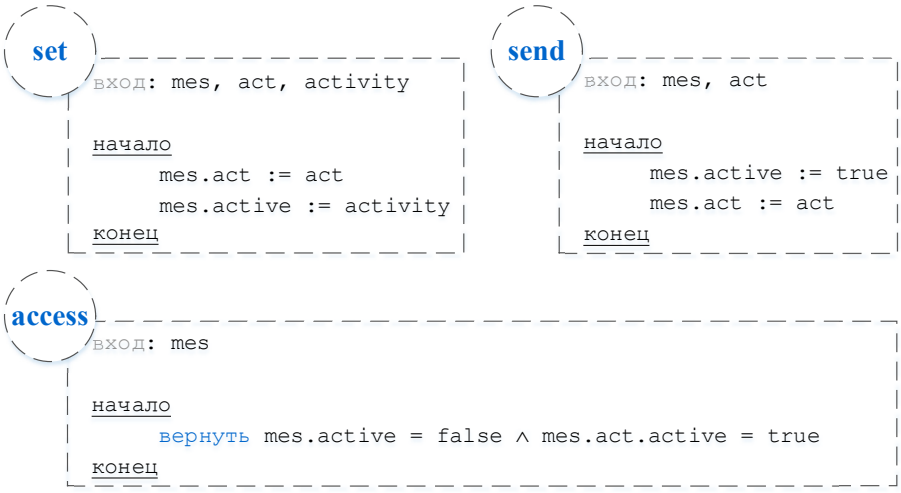

Рисунок 6 – Алгоритмы set, send и access в определении алгоритмического скелета АМ

Согласно спецификации скелета, АМ конкретный алгоритм или производный скелет получается путем определений: init, используя set; recv, используя send и access. Также типы actor и message могут быть расширены путем введения дополнительных информационных полей. Задачей многозадачного информационного процесса в таком определении является любая часть алгоритма recv, которая не содержит обращений к send или access. Параллельно могут выполняться итерации для $\forall m_1, m_2$ в цикле на рисунке 5, если $m_1.active = true \wedge m_2.active = true \wedge m_1.act \neq m_2.act$ Спецификация скелета АМ также предусматривает возможность графического представления информационной структуры созданных на его основе алгоритмов и производных скелетов, предложена форма такого представления.

Метод спецификации информационных процессов применен для построения производных алгоритмических скелетов и алгоритмов разных типов. Схема, отражающая отношения специализации и использования между разработанными в третьей главе алгоритмическими скелетами и алгоритмами предметной области «асинхронные круговые турниры», показана на рисунке 7.

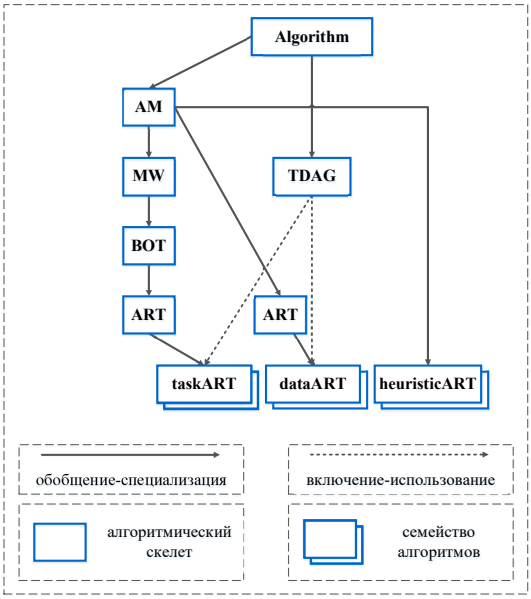

Рисунок 7 – Алгоритмические скелеты и алгоритмы «асинхронных круговых турниров»

В диссертации разработаны следующие алгоритмические скелеты: TDAG – скелет обобщенного алгоритма синтеза графов зависимостей задач из второй главы; AM – базовый скелет акторных алгоритмов; MW – скелет «управляющий-рабочие»; BOT – скелет «портфель задач»; ART – скелет «асинхронного кругового турнира» (рисунок 7). На базе перечисленных скелетов разработано три группы алгоритмов: taskART – «асинхронный круговой турнир» на основе параллелизма задач; dataART – «асинхронный круговой турнир» на основе параллельных потоков данных; heuristicART – алгоритм, реализующий параллельный аналог процесса пузырьковой сортировки.

Вычисления в сети Интернет подразумевают простую и эффективную программную реализацию указанных групп алгоритмов на различных типах современных компьютерных архитектур, представляющих ресурсы гибридного облака. С целью проверки возможности эффективного отображения было разработано и исследовано в нагрузочных тестах 10 программных реализаций «асинхронных круговых турниров». Использовался нагрузочный тест сортировки большого массива. В вычислительных экспериментах проверялись: корректность выполнения; наличие ускорения вычислений; ускорение по сравнению с вариантом чрезвычайно параллельного выполнения, когда только предварительная обработка блоков данных (операции prepare турнира) выполняются параллельно, а игры (операции play турнира) выполняются последовательно; наличие ускорения в результате оптимизации сортирующего турнира; возможность вычислений с мелкогранулярными задачами небольшой длительности; совпадение аналитических оценок сложности вычислений с фактическим поведением информационных процессов. В качестве примера результатов тестирования на рисунках 8 и 9 показаны значения ускорения и абсолютные значения сэкономленного времени в секундах при реализации распределенной сортировки ~80 Гб массива целых чисел на кластере Самарского университета.

Рисунок 8 – Ускорение сортировки в зависимости от числа блоков из 189 * 10^6 целых чисел на кластере Самарского университета (Intel Xeon, MPI, GPFS)

Рисунок 9 – Экономия времени при сортировке в зависимости от числа блоков из 189 * 10^6 целых чисел на кластере Самарского университета

Результаты проведенного тестирования позволяют сделать положительное заключение о работоспособности предложенного в главе метода моделирования информационной структуры многозадачных информационных процессов.

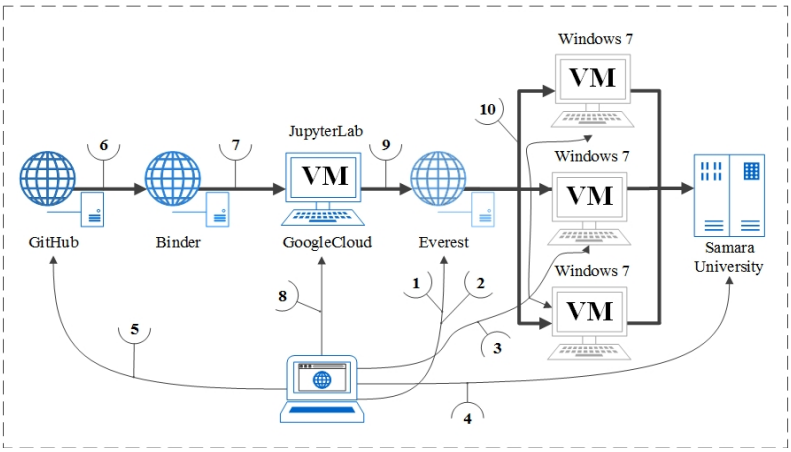

Рисунок 10 – Программная инфраструктура для управления попарной обработкой элементов информационных массивов в гибридном облаке

В четвертой главе предложен общий подход построения программных систем для выполнения попарной обработки больших информационных массивов в гибридном облаке, включающем виртуальные и физические машины гибридного облака, а также бесплатные публичные облачные сервисы. На рисунке 10 показана программная инфраструктура гибридного облака для многозадачной попарной обработки элементов массивов данных и информационное взаимодействие между её компонентами. Цифрами отмечены шаги алгоритма развертывания компонентов. Компоненты инфраструктуры включают: сервис виртуальных рабочих столов Самарского университета (Samara University), предоставляющий доступ к виртуальным машинам (VM) с общей файловой системой; платформу Everest (ИППИ РАН) для реализации взаимодействия между управляющим и вычислительным компонентами и хранения исходного кода вычислительного компонента программной системы;

сервис GitHub для хранения исходного кода управляющего компонента программной системы; сервис Binder для управления развертыванием управляющего компонента программной системы в среде JupyterLab на виртуальной машине в публичном облаке GoogleCloud. В нижней части рисунка 10 показан веб-терминал (веб-браузер), через который пользователь осуществляет настройку системы и управляет вычислениями.

Алгоритм развертывания программной инфраструктуры (по рисунку 10).

Шаг 1. Регистрация вычислительных ресурсов на платформе Everest.

Шаг 2. Установка и настройка компонентов программной системы для выполнения задач prepare и play алгоритмического скелета ART на платформе Everest в виде двух консольных приложений.

Шаг 3. Запуск виртуальных машин для вычислительного компонента программной системы, установка на них программ-агентов вычислительных ресурсов платформы Everest.

Шаг 4. Загрузка обрабатываемого набора данных на файловый сервер, доступный из настроенных на шаге 3 виртуальных машин.

Шаг 5. Запуск управляющей части программной системы, реализующей логику алгоритмического скелета ART, из репозитория на веб-сервисе GitHub.

Шаг 6. Автоматическое обращение к сервису Binder для сборки docker-контейнера с управляющей частью программной системы и средой JupyterLab.

Шаг 7. Развертывание docker-контейнера, созданного на шаге 6, в облаке GoogleCloud. Ссылка на веб-интерфейс управляющей части приложения возвращается в веб-терминал пользователя программной системы.

Шаг 8. Запуск управляющей части приложения пользователем через веб-интерфейс, полученный на шаге 7.

Шаг 9. Проверка состояния запущенных задач и выдача команд запуска очередных задач prepare и play на сервер Everest управляющей частью программной системы.

Шаг 10. Распределение задач prepare и play для выполнения на виртуальные машины вычислительного компонента программной системы сервером платформы Everest через агенты ресурсов, настроенные на шаге 3.

Проведено тестирование разработанной программной инфраструктуры на практической задаче определения частоты повторения слов в понедельных дампах англоязычных сообщений микроблога Twitter по биржевой тематике. Общий объем набора данных в текстовом формате JSON составил 5,88 Гб. Набор разбивался на 10 примерно равных по размеру файлов, обрабатываемых задачами prepare и play на 10 виртуальных машинах в облаке Самарского университета. В задачах prepare выделялось тело сообщения, разбивалось на слова, выполнялось вычисление и упорядочивание пар $\langle$слово, число его повторений$\rangle$, результат записывался в текстовый файл. В задачах play два файла пар $\langle$слово, число его повторений$\rangle$ объединялись в новый упорядоченный массив пар, затем этот массив распределялся на два текстовых файла примерно пополам. В результате запуска задач prepare и play по

алгоритму ART получен упорядоченный набор из 10 файлов пар ⟨слово, число его повторений⟩: слова на букву «A» в первом файле набора, слова на букву «Z» в последнем файле набора. Общее время обработки набора данных составило ~270 секунд, достигнуто 3,6 кратное ускорение.

Главным практическим преимуществом разработанной программной инфраструктуры является возможность обработки данных на произвольных вычислительных ресурсах, например, во время их простоя, а также на арендованных ресурсах в частных или публичных облаках, что позволяет снизить себестоимость и повысить оперативность обработки за счет распараллеливания вычислений. Инфраструктура конфигурируется под различные приложения попарной обработки элементов массивов данных: вычисления корреляций, определения похожих объектов по некоторому критерию, сортировок неструктурированных данных и т.п. Для управления вычислениями пользователю необходимы только сетевое подключение и веб-обозреватель.

В заключении сформулированы основные выводы и результаты, полученные в диссертационной работе.

ОСНОВНЫЕ ВЫВОДЫ И РЕЗУЛЬТАТЫ

1. Выполнен анализ методов, моделей и программных средств, применяемых для глобально распределенной обработки данных. Показана практическая необходимость разработки новых технологий организации вычислений на базе собственных простаивающих ресурсов или ресурсов, арендуемых через сеть Интернет в гибридном облаке. Важный компонент технологий вычислений с использованием указанного типа ресурсов – это специальные многозадачные алгоритмы обработки данных. Перспективным подходом спецификации таких алгоритмов является подход на основе модели акторов Хьюитта и модели алгоритмических скелетов Коула.
2. Разработан метод синтеза алгоритмов асинхронных круговых турниров, выполняющих попарную обработку элементов массивов данных. Метод позволяет синтезировать произвольный многозадачный вычислительный процесс попарной обработки элементов в виде графа зависимостей задач фиксированной структуры. Данный тип процессов является базовой моделью многозадачных вычислений с зависимостями задач по данным и управлению, используемой в приложениях параллельной и распределённой обработки данных.
3. Построено представление трех вычислительных процессов попарной обработки элементов массивов данных в форме графов зависимостей задач. Каждый граф задан порождающим алгоритмом, по которому определена аналитическая оценка ускорения соответствующего вычислительного процесса. При выполнении попарной обработки в турнире без

дополнительных ограничений ускорение равно: $S(M) = \frac{M}{2}$, если M– четное; $S(M) = \frac{M-1}{2}$, если M – нечетное; в простом сортирующем турнире ускорение равно: $S(M) = \frac{M^2-M}{4M-6}$; в оптимизированном сортирующем турнире ускорение равно: $S(M) = \frac{M^2-M}{3M-4}$, $M = 2^k, k \geq 2, k \in N$.

4. Разработана алгоритмическая модель многозадачного вычислительного процесса. Особенностями модели являются описание различных вычислительных процессов на базе общего алгоритмического скелета и определение семантики данного скелета с использованием алгоритмической интерпретации модели акторов Хьюитта, отличающейся от известных функциональных и объектно-ориентированных интерпретаций.
5. Модель представления многозадачного вычислительного процесса применена для реализаций нескольких вариантов многозадачных алгоритмов попарной обработки элементов массивов данных. Показано, что разработанная модель позволяет описывать вычислительные процессы как на основе анализа потоков управления, так и на основе анализа потоков данных, а также строить модели параллельных вычислительных процессов в виде системы алгоритмических скелетов, связанных отношениями обобщение-специализация.
6. Проведено комплексное экспериментальное исследование эффективности программ, построенных на основе алгоритмической модели многозадачного вычислительного процесса в разнотипных вычислительных средах: многоядерном компьютере с общей памятью, высокопроизводительной кластерной системе, сети рабочих станций, гибридном облаке. Эксперименты подтвердили, что унификация структуры алгоритмов упрощает их отображение на разнотипные программно-аппаратные ресурсы гибридного облака, при этом не оказывает существенного негативного влияния на производительность программ.
7. Разработана программная инфраструктура для попарной обработки элементов массивов данных в гибридной облачной среде с использованием собственных простаивающих и/или арендуемых через сеть Интернет-ресурсов. Программная инфраструктура апробирована при решении задачи частотного анализа слов в большом текстовом массиве, построенном на основе алгоритма оптимизированного сортирующего турнира. В вычислительных экспериментах по обработке текстовых данных микроблога Twitter на 10 виртуальных машинах в облаке Самарского университета получено 3,6-кратное ускорение.

Публикации по теме диссертации

Статьи в изданиях, рекомендованных ВАК:

1) Востокин, С.В. Алгоритмы асинхронных круговых турниров для многозадачных приложений обработки данных / С.В. Востокин,

И.В. Бобылева // International Journal of Open Information Technologies. – 2020. – Т. 8. – №. 4. -– С. 45-53.

2) Востокин, С.В. Применение алгоритмических скелетов для проектирования параллельных алгоритмов акторного типа / С.В. Востокин, **И.В. Бобылева** // Международный научный журнал «Современные информационные технологии и ИТ-образование». – 2020. – Т. 16. – №1.

Статьи в изданиях, индексируемых в наукометрической базе Scopus:

1) Vostokin, S.V. Implementing computations with dynamic task dependencies in the desktop grid environment using Everest and Templet Web / S.V. Vostokin, O.V. Sukhoroslov, **I.V. Bobyleva**, S.N. Popov //CEUR Workshop Proceedings. – 2018. – Т. 2267. – С. 271-275.
2) Vostokin, S.V. Implementation of stream processing using the actor formalism for simulation of distributed insertion sort / S.V. Vostokin, **I.V. Kazakova**[1] // Journal of Physics: Conference Series. – 2018. – Т. 1096. – №. 1.
3) Popov, S.N. Distributed block sort: A sample application for data processing in mobile ad hoc networks / S.N. Popov, **I.V. Kazakova**, S.V. Vostokin // International Journal of Innovative Technology and Exploring Engineering. – 2019. – Т. 8. – №. 7. – С. 565-568.
4) Vostokin, S. Building an Algorithmic Skeleton for Block Data Processing on Enterprise Desktop Grids / S. Vostokin, **I. Bobyleva** // Russian Supercomputing Days. – Springer, Cham, 2019. – С. 678-689.
5) Vostokin, S.V. Using the bag-of-tasks model with centralized storage for distributed sorting of large data array / S.V. Vostokin, **I.V. Bobyleva** // CEUR Workshop Proceedings. – 2019. – Т. 2416. – С. 199-203.
6) Vostokin, S. Implementation of frequency analysis of Twitter microblogging in a hybrid cloud based on the Binder, Everest platform and the Samara University virtual desktop service / S. Vostokin, **I. Bobyleva** // CEUR Workshop Proceedings. – 2020. – Т. 2667. – С. 162-165.

Статьи в других изданиях:

1) **Казакова, И.В.** Автоматизация производственных процессов на базе формализма акторов / **И.В. Казакова**, С.В. Востокин // Материалы V Всероссийской научно-технической конференции «Актуальные проблемы ракетно-космической техники» (V Козловские чтения) (11-15 сентября 2017 года, г.Самара). - 2017. - С. 261-264.
2) Востокин, С.В. Реализация потоковых вычислений с использованием формализма акторов для моделирования распределённой сортировки

[1] *Фамилия автора изменена с Казаковой И.В. на Бобылеву И.В. в соответствии со свидетельством о заключении брака II-ЕР № 788841 от 14.04.2018 г.*

вставками / С.В. Востокин, **И.В. Казакова** // Информационные технологии и нанотехнологии (ИТНТ). – 2018. – №. 2018. – С. 2356.

3) **Казакова, И.В.** Микросервисное приложение для распределенной обработки данных на примере задачи блочной сортировки / **И.В. Казакова**, С.Н. Попов, С.В. Востокин // Перспективные информационные технологии (ПИТ 2018). – 2018. – С. 641-643.
4) Востокин, С.В. Метод построения композитных приложений для моделирования динамических систем / С.В. Востокин, **И.В. Казакова**, С.Н. Попов // Управление движением и навигация летательных аппаратов. – 2019. – С. 83-85.
5) Востокин, С.В. Применение модели bag-of-tasks с централизованным хранилищем для распределенной сортировки большого массива данных / С.В. Востокин, **И.В. Бобылева** // Сборник трудов ИТНТ-2019. – 2019. – С. 93-96.
6) **Бобылева, И.В.** Реализация параллельного алгоритма блочной сортировки на основе графа зависимостей задач / **И.В. Бобылева**, С.В. Востокин // Международная молодёжная научная конференция «XV Королёвские чтения», посвящённая 100-летию со дня рождения Д.И. Козлова. – 2019. – С. 466-467.
7) **Бобылева, И.В.** Алгоритмический каркас для блочной обработки данных по принципу кругового турнира / **И.В. Бобылева**, С.В. Востокин // Перспективные информационные технологии (ПИТ 2019). – 2019. – С. 323-325.
8) Востокин, С.В. Реализация частотного анализа микроблога Twitter в гибридном облаке на базе Binder, платформы Everest и сервиса виртуальных рабочих столов Самарского университета / С.В. Востокин, **И.В. Бобылева** // Сборник трудов ИТНТ-2020. – 2020. – С. 64-68.

Свидетельство о регистрации программы для ЭВМ:

1) Востокин, С.В. Программа для управления попарной обработкой элементов информационных массивов в гибридном облаке / С.В. Востокин, **И.В. Бобылева** // Свидетельство о государственной регистрации программы для ЭВМ № 2020663143, выданное Федеральной службой по интеллектуальной собственности. Зарегистрировано в Реестре программ для ЭВМ 22.10.2020.

Бобылева Ирина Владимировна

МЕТОД ПОПАРНОЙ ОБРАБОТКИ ЭЛЕМЕНТОВ ИНФОРМАЦИОННЫХ МАССИВОВ ДЛЯ МНОГОЗАДАЧНЫХ ВЫЧИСЛЕНИЙ В ГИБРИДНОМ ОБЛАКЕ

Автореферат
диссертации на соискание учёной степени
кандидата технических наук

Подписано в печать __.__.2022. Формат бумаги 60×84 1/16.
Усл. печ. л. 1. Заказ ___. Тираж 100 экз.
Отпечатано в типографии АО «РКЦ «Прогресс»
443009, г. Самара, ул. Земеца, 18