

На правах рукописи



Перышкова Евгения Николаевна

**СРЕДСТВА УПРАВЛЕНИЯ РЕСУРСАМИ ВЫЧИСЛИТЕЛЬНЫХ
СИСТЕМ В РЕЖИМЕ ОБСЛУЖИВАНИЯ ПОТОКА ЗАДАЧ
С НЕФИКСИРОВАННЫМИ ПАРАМЕТРАМИ**

Специальность: 05.13.15 – Вычислительные машины,
комплексы и компьютерные сети

Автореферат
диссертации на соискание ученой степени
кандидата технических наук

Новосибирск – 2018

Работа выполнена на Кафедре вычислительных систем федерального государственного бюджетного образовательного учреждения высшего образования «Сибирский государственный университет телекоммуникаций и информатики» (СибГУТИ) Федерального агентства связи.

Научный руководитель:

Курносков Михаил Георгиевич

доктор технических наук, доцент, федеральное государственное бюджетное образовательное учреждение высшего образования «Сибирский государственный университет телекоммуникаций и информатики» (СибГУТИ), заведующий Кафедрой вычислительных систем.

Официальные оппоненты:

Калайда Владимир Тимофеевич

доктор технических наук, профессор, федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский Томский государственный университет» (НИ ТГУ), профессор кафедры оптико-электронных систем и дистанционного зондирования.

Винс Дмитрий Владимирович

кандидат технических наук, федеральное государственное бюджетное учреждение науки Институт вычислительной математики и математической геофизики Сибирского отделения Российской академии наук (ИВМиМГ СО РАН), научный сотрудник лаборатории суперкомпьютерного моделирования.

Ведущая организация – федеральное государственное бюджетное учреждение науки Институт автоматизации и электротехники Сибирского отделения Российской академии наук (ИАиЭ СО РАН)

Защита состоится «27» декабря 2018 г. в 14-00 на заседании диссертационного совета Д 219.005.02 при федеральном государственном бюджетном образовательном учреждении высшего образования «Сибирский государственный университет телекоммуникаций и информатики», по адресу: 630102, г. Новосибирск, ул. Кирова, д. 86, ауд. 625.

С диссертацией можно ознакомиться в библиотеке СибГУТИ и на сайте: http://www.sibsutis.ru/science/postgraduate/dis_sovets/.

Автореферат разослан «__» ноября 2018 г.

Ученый секретарь
диссертационного совета Д 219.005.02
к.т.н., доцент



И.В. Нечта

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность темы исследования и степень ее разработанности. Начиная с середины прошлого столетия в нашей стране и за рубежом ведутся активные исследования в области построения высокопроизводительных вычислительных систем (ВС), которые являются инструментальным средством решения сложных задач, представленных параллельными программами. В архитектурном плане ВС представляет собой совокупность множества элементарных машин (ЭМ) и коммуникационной сети, связывающей их. Основные ресурсы ВС – АЛУ, память и средства ввода/вывода являются логически и технически рассредоточенными по ЭМ. Реализация ЭМ допускает варьирование в широких пределах – от однопроцессорной системы на кристалле до многопроцессорного SMP/NUMA-сервера, оснащенного специализированными ускорителями. Количество ЭМ в современных ВС достигает 98 304 (система IBM BlueGene/Q Sequoia), а число процессорных ядер – 10 649 600 (система Sunway TaihuLight).

Фундаментальный вклад в развитие теории и практики вычислительных систем и средств организации их функционирования внесли выдающиеся ученые, среди которых Е.П. Балашов, В.Б. Бетелин, В.С. Бурцев, В.В. Васильев, В.В. Воеводин, В.М. Глушков, В.Ф. Евдокимов, Э.В. Евреинов, А.В. Забродин, В.П. Иванников, М.Б. Игнатъев, А.В. Каляев, И.А. Каляев, Ю.Г. Косарев, В.В. Корнеев, Л.Н. Королев, А.О. Лацис, С.А. Лебедев, В.К. Левин, И.И. Левин, Г.И. Марчук, В.А. Мельников, Н.Н. Миренков, О.Г. Монахов, Д.А. Поспелов, И.В. Прангишвили, Д.В. Пузанков, Г.Е. Пухов, А.Д. Рычков, Г.Г. Рябов, А.А. Самарский, В.Б. Смоллов, А.Н. Томилин, Я. А. Хетагуров, В.Г. Хорошевский, Б.Н. Четверушкин, Ю.И. Шокин, Н.Н. Яненко, P. Valaji, D. Culler, S. Cray, J. Dongarra, W. Gropp, T. Hoeffler, J.L. Traff, L. Lamport, и др.

Время решения задач на ВС в значительной степени определяется тем, как организовано функционирование системы. Наибольшее распространение получили мультипрограммные режимы функционирования ВС: режимы обслуживания потока и обработки набора задач. В этих режимах на изолированных подсистемах ЭМ системы реализуется множество параллельных программ пользователей. Организация функционирования ВС в мультипрограммных режимах требует решения оптимизационных задач, структура которых определяется информацией, известной о задачах (количестве параллельных ветвей, времени решения, приоритетах задач) и вычислительной системе – принятой модели ВС: допущений относительно топологии коммуникационной сети, конфигурации элементарных машин, а также показателей надежности программно-аппаратных компонентов системы. При решении задач оптимизации функционирования ВС в мультипрограммных режимах большую роль сыграли фундаментальные работы по теории ВС, исследованию операций и оптимальному управлению выдающихся ученых: В.Л. Береснева, Э.Х. Гимади, В.Т. Дементьева, Ю.И. Журавлева, В.С. Танаева, Э.А. Мухачевой, В.Г. Хорошевского, Д.А. Поспелова, D.P. Agrawal, R. Baraglia,

R. Bellman, P. Bouvry, D. Johnson, A. Gara, M. Koffman, R. Perego, K. Steiglitz, H. Taha.

При мультипрограммном функционировании ВС считается, что (В.Г. Хорошевский, 2008), (R. Baraglia, 2008) на ресурсы ВС поступают параллельные задачи, характеризующиеся фиксированным рангом – числом ЭМ, необходимых для их реализации, и временем выполнения. В режиме обработки набора задач, предполагается, что заранее известно количество задач и их параметры. В этом случае требуется построить расписание, определяющее распределение задач по подсистемам ЭМ и последовательность их выполнения. Методы организации функционирования ВС в режиме обработки задач с фиксированными параметрами хорошо развиты и получили распространение в современных системах управления ресурсами ВС (SLURM, TORQUE, Altair PBS Pro, IBM Load Leveler, LSF). Значительная часть систем использует алгоритмы динамического улучшения расписаний путем обратного заполнения окон в них (backfilling). Одним из подходов, позволяющим повысить эффективность эксплуатации ВС, является введение задач с *нефиксированными* параметрами, допускающих запуск на нескольких вариантах конфигураций подсистем ЭМ. Такой подход позволяет алгоритмам организации функционирования ВС находить более эффективные расписания решения задач (по времени выполнения, времени ожидания задач в очереди). Выделяют два типа задач с нефиксированными параметрами (D.G. Feitelson, L. Rudolph, 1996): *пластичные задачи* (moldable) и *адаптирующейся* (malleable, evolving). *Адаптирующиеся задачи* допускают динамическое изменение ранга подсистемы в ходе своего выполнения. На данный момент такие задачи слабо распространены, в мире ведутся работы по созданию средств их реализации (MPI ULFM, Fenix, ReInit; RMix). *Пластичные задачи* допускают решение на подсистемах различных рангов из определенного пользователем множества и не допускают изменения ранга подсистемы в ходе своего выполнения. Такой подход характерен для значительной части MPI-программ, созданных за последние 20 лет. Учитывая сказанное, актуальной является разработка новых моделей, алгоритмов и программного обеспечения организации мультипрограммного обслуживания на ВС потоков задач с нефиксированными параметрами.

В режиме обслуживания потока задач необходимо динамически формировать подсистемы ЭМ различных рангов. При реализации параллельными программами групповых схем информационных обменов (all-to-all, one-to-all scatter, all-to-one gather) возникает одновременное конкурентное использование каналов связи коммуникационной сети ВС. В частности, при заданном размещении процессов по процессорным ядрам, процессы на ядрах одного процессора могут конкурировать за доступ к общему контроллеру памяти; процессы, размещенные на разных процессорах NUMA-узла, разделяют каналы связи межпроцессорного соединения (Intel QPI, HyperTransport); при межузловых обменах процессы одного узла конкурируют за доступ к его коммуникационному контроллеру. Конкуренция за доступ к коммуникационным ресурсам приводит к образованию очередей передачи сообщений в библиотеках стандарта MPI, сетевых адаптерах и коммутаторах,

что увеличивает время реализации информационных обменов. Известные алгоритмы формирования в пределах ВС подсистем ориентированы на системы с определенными структурами сетей межмашинных связей (например, на 2D-решетки, гиперкубические и тороидальные структуры, fat tree, Dragonfly, Kautz и др.) и стремятся сохранять топологическое подобие подсистемы структуре ВС. Применение таких алгоритмов для ВС с иерархической организацией на базе многопроцессорных узлов с общей памятью не обеспечивает предельной эффективности. Остро стоит задача развития известных и создания новых алгоритмов формирования подсистем ЭМ, адекватных шаблонам информационных обменов параллельных программ и учитывающих деградацию производительности каналов связи при их одновременном использовании процессами параллельных программ.

Цель работы и задачи исследования. Целью диссертационной работы является разработка и исследование моделей, алгоритмов и программных средств управления ресурсами вычислительных средств в режиме обслуживания потоков задач с нефиксированными параметрами. В соответствии с целью определены нижеследующие задачи исследования.

1. Выполнить анализ архитектурных свойств современных ВС и подходов к организации функционирования ВС в мультипрограммных режимах.

2. Разработать алгоритмы формирования расписаний решения на ВС параллельных задач с нефиксированными параметрами.

3. Реализовать программные средства поддержки режима обслуживания потока задач с нефиксированными параметрами в системах управления ресурсами TORQUE и Maui.

4. Разработать модель оценки времени реализации коммуникационных операции типа «all-to-all» для ВС на базе многопроцессорных узлов с общей памятью при разных уровнях конкурентного разделения каналов связи процессами параллельных программ.

5. Разработать алгоритм формирования подсистем ЭМ, минимизирующий время реализации коммуникационных операций типа «all-to-all» с учетом конкурентного использования каналов связи процессами параллельных программ.

Научная новизна полученных результатов определяется учетом в созданных методах и алгоритмах нефиксированных параметров параллельных задач, а также иерархической организации коммуникационных сетей ВС и структуры информационных обменов параллельных программ.

1. Новизна разработанных генетических алгоритмов организации мультипрограммного функционирования ВС заключается в возможности задания для каждой параллельной задачи приоритезированного списка конфигурация подсистемы элементарных машин.

2. Предложенные параллельные генетические алгоритмы обработки наборов задач характеризуются линейной масштабируемостью и ориентированы на повышение точности формируемого расписания и сокращения времени его построения.

3. Разработанная модель динамической оценки времени реализации операции типа «all-to-all» в ВС на базе многопроцессорных узлов с общей памятью, в отличие от известных, учитывает падение производительности каналов связи при их конкурентном использовании процессами параллельных MPI-программ.

4. В отличие от известных предложенный алгоритм формирования подсистем элементарных машин выполняет ранжирование и выбор среди всех подсистем заданного ранга оптимальной по времени реализации информационных обменов типа «all-to-all».

Теоретическая и практическая значимость работы состоит в том, что разработанные в диссертации методы и алгоритмы реализованы в виде самостоятельных программных пакетов или расширений систем управления ресурсами ВС, составляющие базу инструментария мультипрограммного обслуживания на ВС потоков параллельных задач с нефиксированными параметрами.

1. Созданные генетические алгоритмы легли в основу пакета MOJOS поддержки мультипрограммных режимов обработки наборов параллельных задач с нефиксированными параметрами. Применение реализованных средств позволяет сократить суммарное время выполнения набора MPI-программ по сравнению с детерминированными алгоритмами FFDH и BFDH. Выполнена интеграция разработанного пакета MOJOS с широко используемыми системами управления ресурсами TORQUE и Maui.

2. Разработанная система оценки времени реализации коллективных операций базируется на предварительном построении таблиц времени выполнения обменов через каналы связи основных функциональных уровней ВС (контролер памяти многоядерного процессора, межпроцессорную шину ЭМ, коммуникационный адаптер ЭМ) при различном числе MPI-процессов, одновременно использующих канал связи. Построение таблиц на уровне MPI позволяет автоматически учесть детали реализации конкретной MPI-библиотеки (порог переключения на протокол для сообщений больших размеров, алгоритм поиска сообщений в очередях запросов и т.д.) и повысить точность прогноза.

3. Программные средства внедрены в действующую мультикластерную ВС Центра параллельных вычислительных технологий СибГУТИ и Лаборатории вычислительных систем Института физики полупроводников им. А.В. Ржанова СО РАН (ИФП СО РАН).

Основные этапы исследования выполнены в рамках проектов №№ 0306-2016-0018, 0306-2018-0012 Президиума РАН; при поддержке грантов Российского фонда фундаментальных исследований №№ 18-41-540005, 16-07-00992, 16-07-00712; 15-07-00048; Совета Президента РФ по поддержке ведущих научных школ РФ № НШ-2175.2012.9 (руководитель – чл.-корр. РАН В.Г. Хорошевский).

Получено четыре свидетельства о государственной регистрации программ для ЭВМ. Результаты работы внедрены в учебный процесс СибГУТИ, в систему параллельного мультипрограммирования пространственно-распределенной ВС

и в систему распределения медиа-контента проекта Сибнет ПАО «Ростелеком», что подтверждается соответствующими актами.

Методология и методы исследования. Для достижения поставленной цели и решения сформулированных в диссертационной работе задач использовались методы теории вычислительных систем и методы локальной оптимизации. Экспериментальные исследования осуществлялись путем моделирования на вычислительных кластерах с иерархической организацией коммуникационных сетей. Работа основана на результатах ведущей научной школы в области анализа и организации функционирования большемасштабных ВС (руководитель – чл.-корр. РАН В.Г. Хорошевский).

Положения и результаты, выносимые на защиту.

1. Генетический алгоритм построения расписаний решения задач с нефиксированными параметрами, обеспечивающий сокращение суммарного времени выполнения задач в среднем на 45 % относительно начальных решений, получаемых известными алгоритмами FFDH и BFDH.

2. Основанные на методе мультистарта параллельные генетические алгоритмы обработки наборов задач с нефиксированными параметрами. Алгоритмы разработаны в модели передачи сообщений и характеризуются линейной зависимостью ускорения от числа процессов.

3. Программное расширение системы управления ресурсами TORQUE, позволяющее сократить время обработки набора задач с нефиксированными параметрами в среднем на 24 % относительно стандартных методов.

4. Программный пакет поддержки режима обслуживания потока задач с нефиксированными параметрами для планировщика Maui, сокращающий время обработки MPI-программ в среднем на 21 % по сравнению с методом «обратного заполнения» (backfilling).

5. Алгоритмические и программные средства оценки времени реализации коммуникационных операции типа «all-to-all», учитывающие падение производительности каналов связи иерархически организованной коммуникационной сети ВС при их конкурентном использовании процессами параллельных MPI-программ.

6. Алгоритм формирования подсистем ЭМ минимизирующий время реализации коллективных обменов типа «all-to-all» и учитывающий загруженность каналов связи, возникающую в следствии их конкурентного использования процессами параллельных программ. На вычислительных кластерах с многопроцессорными NUMA-узлами и сетью связи стандарта InfiniBand алгоритм обеспечивает сокращение времени информационных обменов от 16% до 31% по сравнению с известным алгоритмом FF (first fit).

7. Развитая конфигурация мультикластерной ВС и инструментарий параллельного мультипрограммирования, расширенные пакеты обслуживания потоков параллельных задач с нефиксированными параметрами.

Личный вклад. Выносимые на защиту результаты получены соискателем лично. В совместных работах постановки задач и разработка методов их решения осуществлялись при непосредственном участии соискателя: разработка и исследование генетических алгоритмов построения расписаний решения задач с

нефиксированными параметрами [1–3], программная реализация алгоритмов и их экспериментальное исследование [4, 7–9].

Степень достоверности и апробация результатов подтверждаются проведенными экспериментами и моделированием, согласованностью с данными, имеющимися в отечественной и зарубежной литературе. Основные результаты работы докладывались и обсуждались на международных, всероссийских и региональных научных конференциях, в их числе: международные конференции «Актуальные проблемы электронного приборостроения» (г. Новосибирск, 2018), Российская конференции с международным участием «Новые информационные технологии в исследовании сложных структур» (г. Томск, 2018), «Визуальная аналитика» (г. Кемерово, 2017), школа-семинар «Проблемы оптимизации сложных систем» в рамках международной конференции IEEE SIBIRCON-2017 (г. Новосибирск, 2017), Всероссийская научно-практической конференции с международным участием «Интеллектуальный анализ сигналов, данных и знаний: методы и средства» (г. Новосибирск, 2017), «Инжиниринг & Телекоммуникации» (г. Москва, 2015), российские конференции: «Обработка информации и математическое моделирование» (г. Новосибирск, 2018) «Национальный суперкомпьютерный форум» (г. Переславль-Залесский, 2016), «Перспективные информационные и телекоммуникационные технологии» (г. Новосибирск, 2016).

Публикации. По теме диссертации опубликована 21 работа. Из них: 4 – в журналах из перечня ВАК РФ; 1 – в изданиях, индексируемых Scopus. Получено 4 свидетельства о государственной регистрации программ для ЭВМ.

Структура и объем диссертации. Диссертационная работа состоит из введения, четырех глав, заключения, списка литературы и приложений, изложенных на 148 страницах.

ОСНОВНОЕ СОДЕРЖАНИЕ РАБОТЫ

Во введении изложена актуальность темы исследования и степень ее разработанности, сформулированы цели и задачи работы, отражена ее научная новизна, теоретическая и практическая значимость; представлены положения, выносимые на защиту, а также степень достоверности и апробация результатов.

В первой главе дается понятие о вычислительных системах, рассматриваются особенности технической организации кластерных вычислительных систем и вычислительных систем с массовым параллелизмом. Описаны режимы функционирования вычислительных систем и программно-аппаратный стек технологий для организации работы таких систем.

В мультипрограммных режимах работы ВС на изолированных подсистемах ЭМ реализуется множество параллельных программ пользователей. В режиме обработки набора задач, предполагается, что заранее известно количество задач и их параметры: число требуемых ЭМ и ожидаемое время выполнение. В этом случае требуется построить расписание, определяющее распределение задач по ЭМ системы и моменты запуска задач на них. В режиме обслуживания потока

задач их количество заранее неизвестно. Здесь необходимо для поступающих задач динамически формировать оптимальные подсистемы ЭМ.

Алгоритмы параллельного мультипрограммирования являются основой систем управления ресурсами (СУР) ВС (например, SLURM, TORQUE, Altair PBS Pro, IBM Load Leveler, LSF). На примере системы управления ресурсами SLURM показано, что сложность цикла планирования СУР характеризуется линейно-логарифмической зависимостью числа операций от количества задач, поступающих на выполнение, и количества ЭМ в распределенной ВС.

От эффективности математических моделей и алгоритмов организации мультипрограммной работы ВС зависит эффективность ее функционирования: время решения задач, сбалансированность загруженности вычислительных ресурсов, энергопотребление. Остро стоит проблема развития известных и создания новых моделей, методов и алгоритмов организации мультипрограммного функционирования ВС адекватных современному уровню архитектуры ВС.

Вторая глава посвящена организации мультипрограммного функционирования ВС в режиме обработки наборов задач с нефиксированными параметрами.

В классической постановке (В.Г. Хорошевский, 1967, 2008; Д.А. Поспелов, 1972; Танаев В.С., 1975; E. Coffman, 1978, P. Brucker, 2006), организация мультипрограммной обработки набора задач подразумевает построение (суб)оптимального расписания решения на элементарных машинах ВС задач с фиксированными параметрами – рангом (числом требуемых машин) и временем решения. В качестве критерия оптимизации используется суммарное время решения всех задач. Данная проблема относится к классу трудноразрешимых. С середины 1960-х годов разработаны точные и приближенные методы обработки набора задач на ВС. Точные методы основаны на полном переборе множества допустимых расписаний. Здесь широкое распространение получил направленный перебор методом ветвей и границ. В силу высокой вычислительной сложности точные методы крайне редко применяются на практике для оперативной обработки наборов задач.

В большей части работ по параллельному мультипрограммированию исходная проблема обработки набора задач (ПОНЗ) решается путем полиномиального сведения к решению ряда других оптимизационных задач. Например, в первых работах по параллельному мультипрограммированию ПОНЗ сводилась к последовательному решению двух задач одномерной упаковки объектов в контейнеры (1D Bin Packing Problem, 1DBPP). При этом задача 1DBPP решается уже известными стохастическими алгоритмами локального поиска или детерминированными эвристиками FFD, BFD с гарантированной оценкой точности. В ряде других работ используется сведение ПОНЗ к задаче двумерной упаковки прямоугольников (параллельных задач) в полуограниченную полосу (2D Strip Packing Problem, 2DSPP). Для решения 2DSPP применяют алгоритмы с гарантированной оценкой точности FFDH, NFDH, BFDH, а также стохастические алгоритмы локального поиска:

генетические алгоритмы, имитации отжига (simulated annealing), поиск с запретами (tabu search), муравьиной колонии (ant colony).

Активно развиваются и методы динамического улучшения уже построенных расписаний, в частности, широкое распространение в СУР получили методы «обратного заполнения» (backfilling) свободных окон в расписании.

Одним из подходов, позволяющим повысить эффективность эксплуатации ВС в мультипрограммном режиме, является введение задач с *нефиксированными параметрами* (пластичные задачи, moldable job), допускающих запуск на нескольких вариантах конфигураций подсистем ЭМ. Такой подход позволяет алгоритмам организации функционирования ВС находить более эффективные расписания решения задач на ВС.

Рассмотрим пример набора из 5 параллельных задач с фиксированными параметрами:

$$p_1 = (2, 7), \quad p_2 = (5, 4), \quad p_3 = (3, 2), \quad p_4 = (7, 3), \quad p_5 = (4, 4).$$

Первый параметр характеризует ранг задачи, второй – время ее решения. Представим каждую задачу в виде прямоугольника с шириной, равной количеству требуемых ЭМ, и высотой, равной времени решения задачи. На рисунке 1.а показан пример допустимого расписания. Заполнить образовавшиеся в расписании свободные «окна» можно разными методами. Например, если бы задача 4 допускала запуск не только на подсистемах ранга 7, но и на других вариантах подсистем (имела вектор допустимых параметров), то алгоритм планирования имел бы больше вариантов для построения «плотного расписания». На рисунке 1.б показан пример расписания, в котором задача 4 имеет нефиксированные параметры $p_4 = \langle (7, 3), (4, 5) \rangle$ и допускает запуск на подсистеме из 4 ЭМ. Это позволило сократить суммарное время решения задач с 11 до 9 единиц времени.

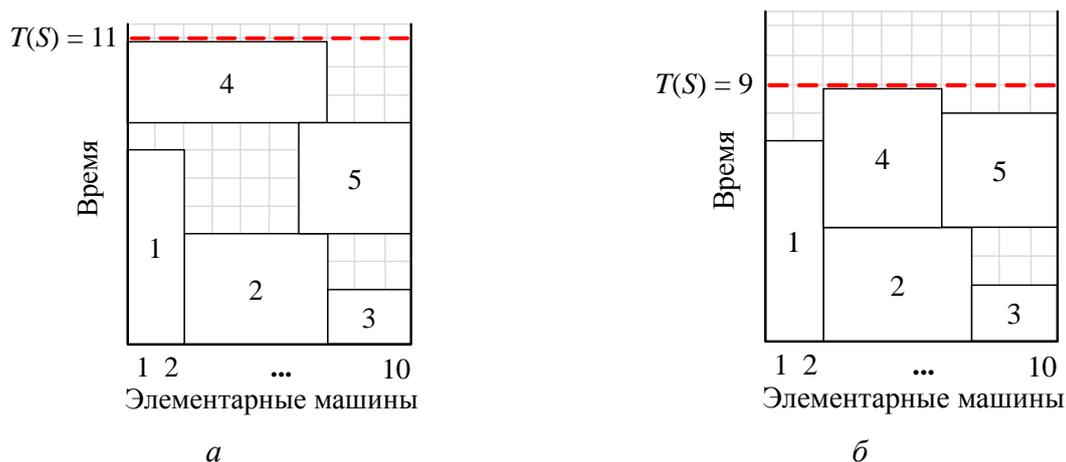


Рис. 1 Расписание решения 5 задач на распределенной ВС из 10 ЭМ:
 а) набора задач с фиксированными параметрами: время решения $T(S) = 11$;
 б) модифицированный набор с задачей $p_4 = \langle (7, 3), (4, 5) \rangle$: $T(S) = 9$

Введение задач с нефиксированными параметрами обусловлено не только необходимостью повысить эффективность формируемых расписаний, но и продиктовано необходимостью учитывать современные технико-экономические ограничения:

– некоторые параллельные задачи допускают реализацию только на подсистемах ЭМ с определенными свойствами, например, с числом процессорных ядер равным степени числа два;

– лицензии значительной часть коммерческих пакетов параллельного моделирования допускают их запуском на фиксированных конфигурациях подсистем ЭМ;

– отказы ресурсов в большемасштабных ВС и перспективных системах экзафлопсной производительности требуют поддержки возможность восстановления вычислительного процесса на допустимой подсистеме меньшего ранга.

Проблема обработки набор задач с нефиксированными параметрами относится к дискретной оптимизации и является трудноразрешимой. Имеется распределенная ВС, состоящая из n элементарных машин, и набор из m параллельных задач с нефиксированными параметрами. Для каждой параллельной задачи $i \in \{1, 2, \dots, m\}$ задан вектор $p_i = (p_i^1, p_i^2, \dots, p_i^{q_i})$ из q_i различных вариантов параметров задачи (вектор допустимых подсистем). Каждый элемент $p_i^k = (r_i^k, t_i^k, w_i^k)$ описывает допустимый вариант параметров: t_i^k – время решения задачи на подсистема ранга r_i^k , а w_i^k – заданный пользователем приоритет (предпочтение) выбора параметра $k \in \{1, 2, \dots, q_i\}$ при формировании расписания. Допускается существование в векторе p_i нескольких элементов с одинаковым приоритетом. Требуется в условиях ограниченности ресурсов ВС составить расписание S решения задач:

$$S = ((s_1, x_1, k_1), (s_2, x_2, k_2), \dots, (s_m, x_m, k_m)).$$

здесь s_i – время начала решения i -ой задачи на ВС, а $x_i = (x_{i1}, x_{i2}, \dots, x_{ir_i})$ – подсистема ЭМ для выполнения программы, $x_{ij} \in C$ – номер ЭМ для выполнения ветви j задачи i . Также необходимо определить $k_i \in \{1, 2, \dots, q_i\}$ – номер выбранного варианта параметров задачи i .

Обозначим через u_i^k «удовлетворенность» пользователя выбором для решения задачи i параметров с номером $k \in \{1, 2, \dots, q_i\}$

$$u_i^k = \frac{w_i^k}{\max_{k=1, q_i} w_i^k}.$$

Формируемое расписание должно обеспечивать среднюю «удовлетворённость» пользователей не ниже заданного значения w :

$$\frac{1}{m} \sum_{i=1}^m \frac{w_i^{k_i}}{\max_{k=1, q_i} w_i^k} \geq w.$$

Обозначим через $J(t) = \{i \in J \mid s_i \leq t \leq s_i + t_i\}$ множество задач, решаемых на ВС в момент времени t , а через Ω множество допустимых расписаний. В качестве показателя оптимальности расписаний используется время $T(S)$ окончания решения последней задачи:

$$T(S) = \max_{i \in J} \{s_i + t_i\}.$$

Итак, требуется найти допустимое расписание $S \in \Omega$, доставляющее минимум целевой функции $T(S)$. Формально

$$T(S) = \max_{i \in J} \{s_i + t_i\} \rightarrow \min_{S \in \Omega} \quad (1)$$

при ограничениях:

$$\sum_{i \in J(t)} r_i \leq n, \quad \forall t \in \mathbb{R}, \quad (2)$$

$$\prod_{i \in J(t)} \prod_{i' \in J(t) \setminus \{i\}} (x_{ij} - x_{i'j'}), \quad \forall t \in \mathbb{R}, \quad j = 1, 2, \dots, r_i, \quad j' = 1, 2, \dots, r_{i'}, \quad (3)$$

$$\frac{1}{m} \sum_{i=1}^m \frac{w_i^{k_i}}{\max_{k=1, \dots, q_i} w_i^k} \geq w. \quad (4)$$

$$x_{ij} \in C, \quad i = 1, 2, \dots, m, \quad j = 1, 2, \dots, r_i^{k_i}, \quad (5)$$

$$s_i \in \mathbb{R}, \quad k_i \in \{1, 2, \dots, q_i\}, \quad i = 1, 2, \dots, m. \quad (6)$$

Ограничения (2) и (3) гарантируют, что в любой момент времени для реализации ветвей задач используются различные ЭМ, количество которых не превосходит n . Для приближенного решения задачи (1) – (6) автором разработаны последовательные и параллельные генетические алгоритмы с полиномиальной вычислительной сложностью. Алгоритмы относятся к классу стохастических алгоритмов локальной оптимизации.

Генетический алгоритм формирования расписания решения задач с нефиксированными параметрами. Идея алгоритма заключается в создании и эволюции популяции из допустимых расписаний решения задач с нефиксированными параметрами. Алгоритм состоит из 5 шагов.

Шаг 1. Создание начальной популяции из K допустимых расписаний. При формировании начальной популяции параметр k_i задачи i выбирается псевдослучайным образом из множества $\{1, 2, \dots, q_i\}$. Расписание формируется с помощью эвристических алгоритмов FFDH или VFDH. Если полученное расписание удовлетворяет ограничениям (2) – (4), оно сохраняется в популяции. Данный шаг повторяется, пока не будет получено K допустимых расписаний.

Шаг 2. Определение двух особей (расписания) для скрещивания и получения двух новых расписаний. Получение новых особей осуществляется с помощью многоточечного оператора скрещивания – каждое расписание закодированные в векторе псевдослучайным образом делится на G частей. Далее, выбранная пара особей обменивается полученными частями. Если особь не скрещивается, к ней применяется оператор мутации, который псевдослучайным образом изменяет номер k_i выбранного параметра задачи.

Шаг 3. Упорядочивание всей популяции по значению целевой функции $T(S)$. В популяции остаются лучшие K особей (расписаний), остальные удаляются.

Шаг 4. Проверка окончания эволюции. Если количество эволюционных циклов не достигло предельного значения V , то возврат к шагу 2.

Шаг 5. Выбор итогового решения. За итоговое решение принимается особь с экстремальным значением целевой функции $T(S)$ в текущей популяции.

Сформулировано и доказано утверждение о вычислительной сложности

предложенного алгоритма. Его трудоемкость составляет:

$$T = O(Km + KT_{2DSPP} + VK \log K + VK),$$

где T_{2DSPP} – вычислительная сложность формирования упаковки прямоугольников в полуограниченную полосу (время работы алгоритма FFDH или BFDH), K – размер популяции, V – число эволюционных циклов.

Проведено моделирование работы алгоритма на тестовых наборах задач с нефиксированными параметрами с числом ЭМ $n = 1024, 4096, 16384, 65536$ и количеством задач в наборе $m = 1000, 2000$ и 3000 . В качестве оценок эффективности (точности) работы алгоритма рассматривались величины:

$$\delta_1 = (T_{BFDH} - T)/T, \delta_2 = (T_{FFDH} - T)/T,$$

$$\delta_3 = (T_{INIT_BFDH} - T)/T, \delta_4 = (T_{INIT_FFDH} - T)/T,$$

где T_{INIT_BFDH} – значение целевой функции от начального решения, полученного с помощью алгоритма упаковки BFDH, T_{INIT_FFDH} – значение целевой функции от начального решения, полученного с помощью алгоритма упаковки FFDH, T – значение целевой функции от решения, получаемого генетическим алгоритмом, а T_{BFDH} и T_{FFDH} – значение целевой функции от решения, получаемого алгоритмом упаковки BFDH и FFDH, соответственно.

Для оценки эффективности работы алгоритма производилось 100 запусков для набора задач каждого из указанных размеров. В среднем генетический алгоритм на рассмотренных наборах модельных данных позволяет получать решение задачи на 127 % лучше алгоритмов упаковки и на 45 % лучше начального решения.

Качество получаемого расписания зависит от выбранных параметров генетического алгоритма. На рис. 2.а и рис. 2.б приведены графики зависимости значений оценок математических ожиданий $M[\delta_1]$, $M[\delta_2]$, $M[\delta_3]$ и $M[\delta_4]$ от количества эволюционных циклов V для наборов с количеством задач $m = 1000$ и размером системы $n = 1024$.

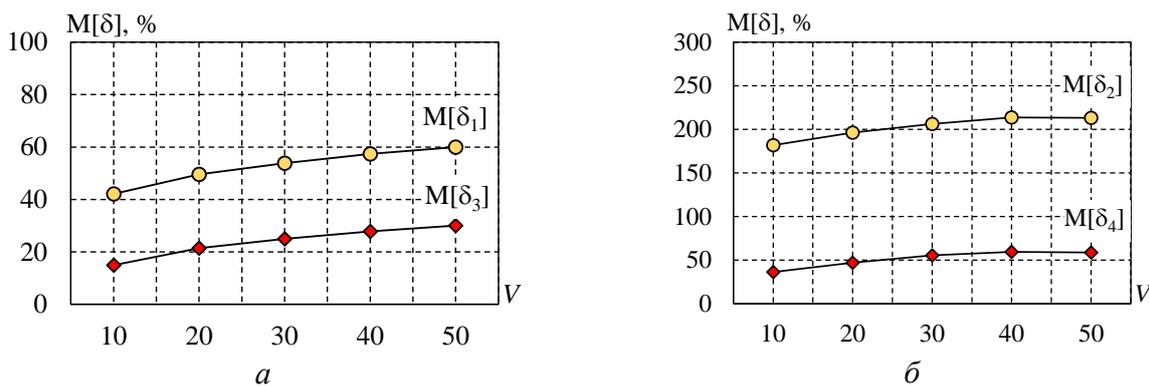


Рис. 2. Зависимость математического ожидания от количества эволюционных циклов V ($n = 1024, m = 1000; K = 16, w = 75 \%, P = 90 \%, G = 2$):

а) оценки $M[\delta_1]$ и $M[\delta_3]$; б) оценки $M[\delta_2]$ и $M[\delta_4]$

Зависимости математического ожидания и среднеквадратического отклонения для δ_1 и δ_2 от количества эволюционных циклов V показаны на рис. 3.а и 3.б.

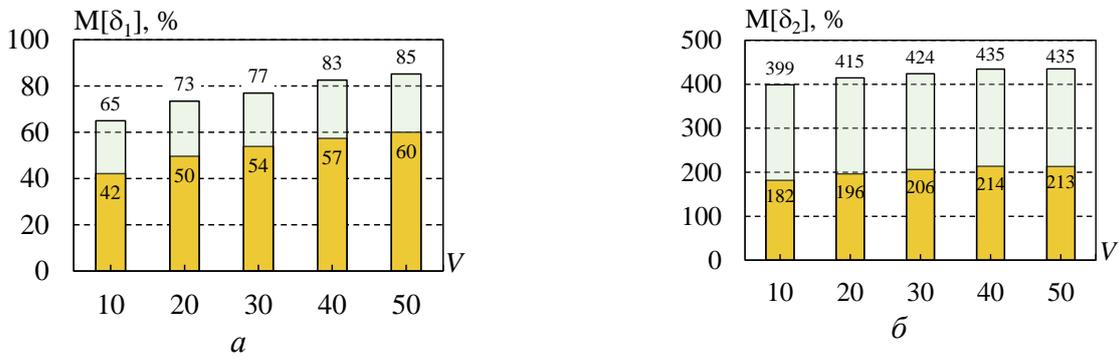


Рис. 3. Зависимость математического ожидания и среднеквадратического отклонения от количества эволюционных циклов V ($n = 1024, m = 1000; K = 16, w = 75 \%, P = 90 \%, G = 2$):
 а) для δ_1 ; б) для δ_2

На рис. 4 приведены зависимости времени работы генетического алгоритма построения расписания решения наборов, состоявших из $m = 1000, 2000$ и 3000 задач для системы из $n = 1024$ ЭМ. В качестве алгоритма формирования начальной популяции использовались алгоритмы упаковки прямоугольников в полуограниченную полосу BFDH (рис. 4.а) и FFDH (рис. 4.б).

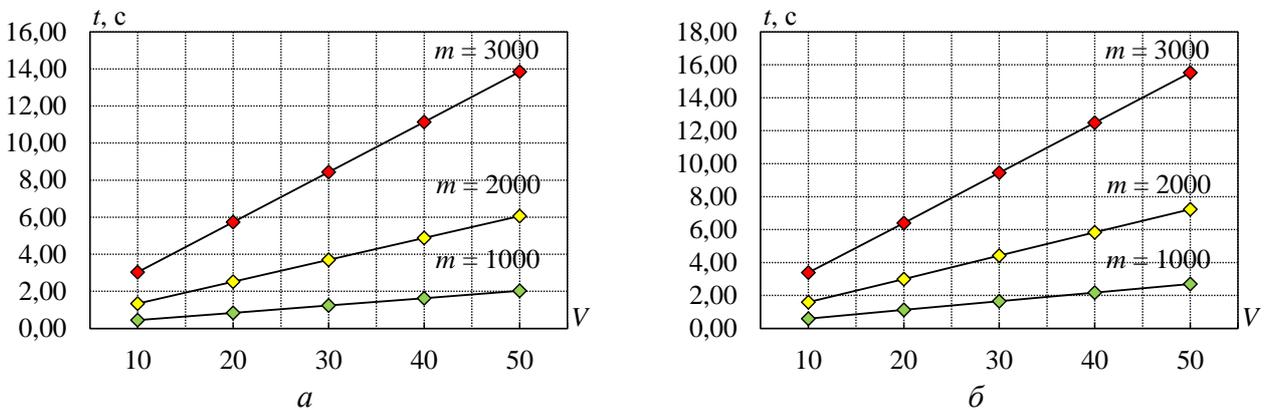


Рис. 4. Зависимость времени работы алгоритма от количества эволюционных циклов V и числа m задач в наборе для ВС из 1024 ЭМ:
 а) алгоритм BFDH; б) алгоритм FFDH

Для обработки больших наборов задач автором созданы параллельные генетические алгоритмы, основанные на метода мультистарта – параллельного поиска эффективных решений из разных начальных точек множества допустимых расписаний. Алгоритмы реализованы на языке С в модели передачи сообщений. Эксперименты на вычислительных кластерах показали их масштабируемость близкую к линейной.

Созданные генетические алгоритмы легли в основу программного пакета MOJOS поддержки мультипрограммных режимов обработки наборов параллельных задач с нефиксированными параметрами. Пакет реализован на языке С++ и поддерживает интеграцию с системой управлениями ресурсами TORQUE и планировщиком Maui.

Для интеграции поддержки режима обслуживания потока задач с нефиксированными параметрами язык описания ресурсных запросов системы TORQUE расширен структурой $ATTR_L$ «*Moldable_Resource_List*» для описания

вектора допустимых параметров. Вектор допустимых подсистем имеет следующий вид:

$-L \text{ nodes} = \text{value} @ \text{ppn} = \text{value} @ \text{weight} = \text{value} @ \text{walltime} = \text{value}$,
 где nodes – число требуемых ЭМ (обязательный параметр); ppn – число процессорных ядер на каждом узле; weight – приоритет данного запроса относительно других запросов (натуральное число); walltime – максимальное время использования ресурсов.

Проведено экспериментальное исследование эффективности расширения СУР TORQUE и планировщика Maui для управления потоком параллельных задач с нефиксированными параметрами на вычислительном кластере с двухпроцессорными вычислительными узлами и сетью связи стандарта Gigabit Ethernet. Генерировались тестовые наборы с числом MPI-задач $m = 100, 200, 400$ и 800 для обслуживания на вычислительном кластере из 18 двухпроцессорных узлов (144 процессорных ядра). Запуск осуществлялся 100 раз для наборов каждого размера. В качестве оценок эффективности рассматривались величины:

$$\delta_1 = (T_T - T)/T, \delta_2 = (T_M - T)/T, \delta_3 = (T_{TQ} - T_Q)/T_Q, \delta_4 = (T_{MQ} - T_Q)/T_Q,$$

где T_T и T_{TQ} – время решения задач набора и время ожидания задач в очереди, полученное с помощью СУР TORQUE, T_M и T_{MQ} – время решения задач набора и время ожидания задач в очереди, полученное при использовании планировщика Maui, T и T_Q – время решения задач набора и время ожидания задач в очереди, полученное при интеграции созданного программного инструментария. В среднем применение разработанных средств для системы TORQUE на рассмотренных наборах позволяет сократить суммарное время решения зада на 24 % и на 21 % для планировщика Maui. В табл. 1 и табл. 2 представлены время выполнения и времени ожидания в очереди задач для СУР TORQUE и планировщика Maui.

Табл. 1. Суммарное время обработки набора задач

Количество задач в наборе	$T_T, \text{с}$	$T_M, \text{с}$	$T, \text{с}$	$M[\delta_1], \%$	$M[\delta_2], \%$
100	13403	12317	11188	19,80	10,09
200	50426	45171	43195	16,74	4,57
400	207528	202364	175033	18,57	15,61
800	836394	796453	713580	17,21	11,61

Табл. 2. Время ожидания задач в очереди

Количество задач в наборе	$T_{TQ}, \text{с}$	$T_{MQ}, \text{с}$	$T_Q, \text{с}$	$M[\delta_3], \%$	$M[\delta_4], \%$
100	323	310	246	31,30	26,02
200	372	356	326	14,11	9,20
400	684	697	605	13,06	15,21
800	1324	1294	1200	10,33	7,83

Третья глава посвящена алгоритмам формирования подсистем ЭМ для выполнения параллельных программ. Предложен эвристический алгоритм формирования подсистем ЭМ с учетом возможной деградации

производительности каналов межмашинных связей при их одновременном (конкурентном) использовании процессами параллельных программ.

Задача формирования подсистем ЭМ возникает при функционировании ВС в мультипрограммном режиме обслуживание потока задач, представленных параллельными программами. Требуется для каждой задачи определить подсистемы ЭМ требуемого ранга (количество процессорных ядер, необходимых для выполнения параллельной программы) и распределить по ним ветви параллельной программы. Системы управления ресурсами ВС должны сформировать подсистемы ЭМ, обеспечивающие экстремальное значение показателя эффективности. На практике широкое распространение получили следующие показатели эффективности подсистем ЭМ: средний диаметр подсистемы, среднее значение пропускной способности каналов связи между ЭМ подсистемы, вектор-функция структурной коммутационности и др. Подавляющее большинство современных ВС строится на базе многопроцессорных SMP/NUMA-узлов с общей памятью. Особенностью формирования подсистем ЭМ в таких ВС является то, что подсистема заданного ранга может быть сформирована разными способами. Например, симметричная подсистема с количеством ЭМ (процессорных ядер), равном восьми, может быть сформирована тремя способами: один вычислительный узел с 8 процессорными ядрами (1x8), два узла по 4 ядра (2x4) и четыре узла по 2 ядра (4x2). Время выполнения коммуникационных операций на этих подсистемах будет различным в силу иерархической организации коммуникационной сети ВС. Системе управления ресурсами необходимо выбрать из данного множества допустимых подсистем одну, обеспечивающую минимум времени реализации информационных обменов. В диссертации предложен алгоритм формирования подсистем ЭМ, минимизирующий время реализации коллективных обменов типа «all-to-all», и учитывающий загруженность каналов связи, возникающую в следствии их конкурентного использования процессами параллельных программ. Ориентация алгоритма на коллективную операцию «all-to-all» обусловлена отсутствием у СУР априорной информации о шаблоне информационных обменов в параллельной программе. Производительность этой операции является критически важной для широкого спектра параллельных программ, в их числе, MPI-реализации алгоритмов на графах (Graph500), быстрого преобразования Фурье (НРСС FFT), параллельные реализации методов решения из первых принципов (ab initio) задач квантовой химии (Quantum Espresso) и др.

Алгоритм формирования подсистем ЭМ. Имеется распределенная ВС, состоящая из n многопроцессорных элементарных машин. Из доступных ЭМ может быть сконфигурировано k различных симметричных подсистем ранга r . Ранг подсистемы r – суммарное количество логических процессоров в ней (процессорных ядер). Каждая подсистема задана парой (n_i, p_i) , где n_i – число вычислительных узлов в подсистеме, p_i – число логических процессоров, на каждом узле. Требуется из k подсистем выбрать одну, обеспечивающую минимум времени выполнения коллективной операции «all-to-all» (MPI_Alltoall). Размер передаваемых сообщений не задан, но допускается

предположение о классе размера сообщений: короткие (до 1 Кбайт), средние (до 128 Кбайт) и большие (свыше 128 Кбайт). В общем случае требуется установить отношение $\tau(s_1, s_2)$ частичного порядка на множестве $S = \{1, 2, \dots, m\}$ подсистем ранга r : $\tau(s_1, s_2)$: «время реализации обмена all-to-all на подсистеме s_1 не превосходит время реализации на подсистеме s_2 ».

Предложенный алгоритм включает два этапа:

Шаг 1 (выполняется при начальной установке и настройке СУР). Оценка разработанными тестами производительности каналов связи ВС при различном уровне их конкурентно использования MPI-процессами.

Шаг 2 (выполняется при поступлении задачи). Динамическое оценка на модели времени реализации операции «all-to-all» на доступных подсистемах ранга r . Выбор подсистем исходя из минимального оценочного времени реализации коллективной операции «all-to-all».

В диссертации разработана *система динамической оценки времени выполнения информационных обменов*, учитывающий загруженность каналов связи, возникающую в следствии их конкурентного использования процессами параллельных программ. Функциональная структура системы оценки представлена на рис. 5.

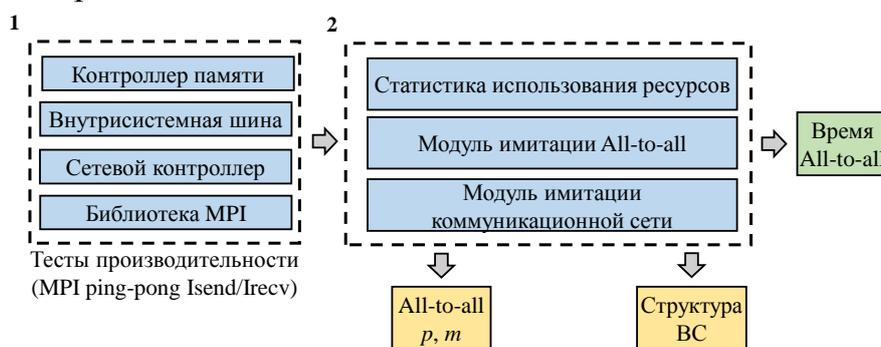


Рис. 5. Функциональная структура системы оценки времени выполнения коллективной операции «all-to-all» на заданной подсистеме ЭМ

Первый шаг реализуется *Модулем 1* для измерения времени выполнения операции обмена при различных уровнях конкурентного разделения логических каналов связи. При реализации параллельными программами глобальных схем информационным обменом возникает одновременное совместное использование каналов связи (*network contention*). Следствием этого является образование очередей передачи сообщений в библиотеках стандарта MPI, сетевых адаптерах, коммутаторах и падение производительности коммуникационной сети. Множество используемых при передаче сообщений между параллельными процессами MPI-программ каналов связи определяется начальным распределением процессов по процессорным ядрам ЭМ системы. Например, на рис. 6.а показан пример взаимодействия MPI-процессов, размещенных на двух ядрах одного процессора SMP-узла. В этом случае обмен осуществляется через оперативную память узла. В аналогичной ситуации для NUMA-узлов два процесса выполняют обмен через оперативную память NUMA-узла, на ядрах процессора которого они выполняются. При взаимодействии ядер, размещенных на разных процессорах NUMA-узла, сообщения передаются через

внутрисистемную шину, например, Intel QuickPath Interconnect (QPI), как показано на рис. 6.б. Если взаимодействующие ядра размещены на процессорах, находящихся на разных ЭМ, обмен осуществляется через сетевой адаптер, как показано на рис. 6.в.

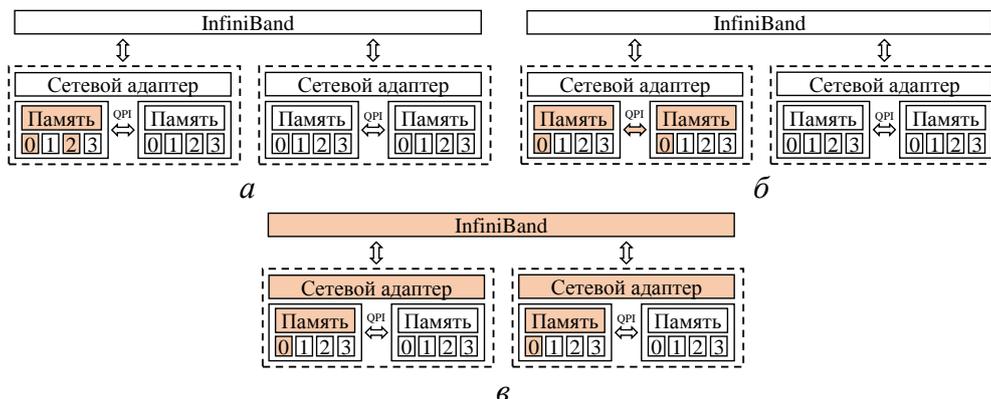


Рис. 6. Возникновение конкуренции за разделяемые ресурсы:

- а) контроллер памяти (процессы привязаны к ядрам разных процессоров);
- б) шина Intel QPI (процессы привязаны к процессорным ядрам разных процессоров, соединенных шиной Intel QPI);
- в) сетевой адаптер (процессы привязаны к процессорным ядрам на разных ЭМ);

Для определения влияния конкуренции за сетевые ресурсы на время выполнения информационных обменов выполняется запуск тестовой MPI-программы, оценивающей время выполнения дифференциальных обменов через каналы связи основных функциональных уровней ВС при различных уровнях их конкурентного разделения процессами параллельной программы. На рис. 7-9 показаны зависимости $t(m, cf)$ времени передачи сообщения размером m байт от количества cf (contention factor) процессов, одновременно разделяющих общий канал связи.

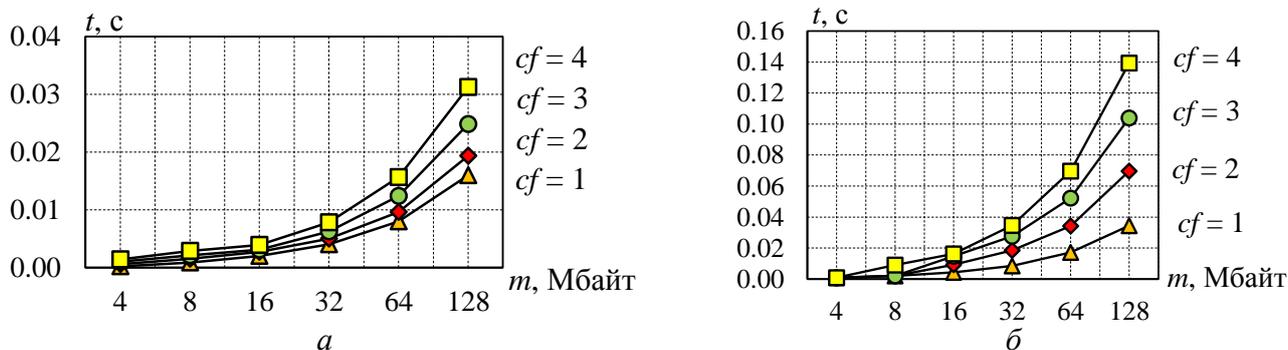


Рис. 7. Измерение времени t передачи сообщения m и количества cf процессов MPI-программы, разделяющих общий канал связи:

- а) контроллер памяти NUMA-узла (два процесса привязаны к двум ядрам одного процессора с интегрированным контроллером памяти);
- б) контроллер памяти SMP-узла (два процесса привязаны к двум ядрам разных процессоров);

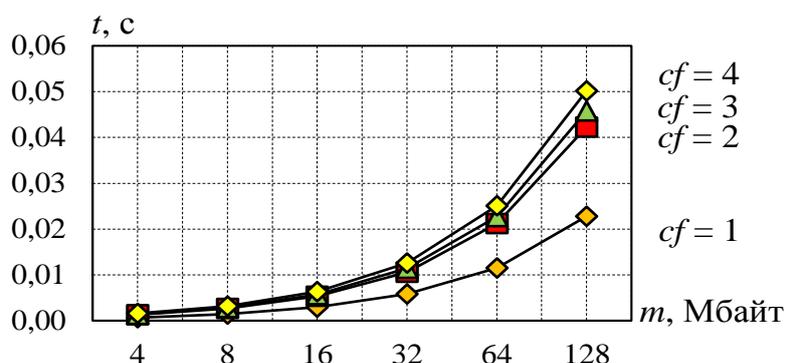


Рис. 8. Измерение времени t передачи сообщения m и количества cf процессов MPI-программы, разделяющих внутрисистемную шину Intel QPI (два процесса привязаны к процессорным ядрам разных процессоров, соединенных шиной Intel QPI)

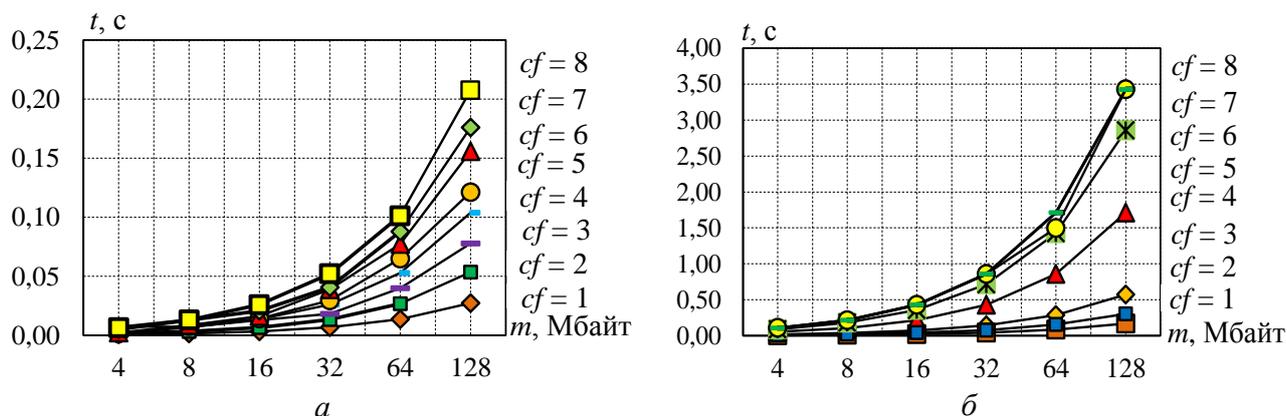


Рис. 9. Измерение времени t передачи сообщения m и количества cf процессов MPI-программы, разделяющих общий канал связи:

а) сетевой контроллер InfiniBand; б) сетевой контроллер Gigabit Ethernet;

Экспериментальное построение таблично-заданные функции $t(l, m, cf)$ времени выполнения операции передачи сообщения размером m байт при разделении cf процессами канала связи на уровне l производится один раз на заданной системе и, в дальнейшем, используются для динамического построения оценок времени реализации алгоритмов.

На втором шаге работы алгоритмы используется *Модуль 2*, включающий модуль сбора статистики одновременного использования ресурсов, модуль имитации блочного алгоритма «all-to-all» и модуль имитации иерархической коммуникационной сети. Модуль сбора статистики предназначен для подсчета числа cf одновременных использований каналов связи при реализации одного шага конкретного алгоритма (шаблона) информационных обменов параллельной программы.

Модуль имитации иерархической коммуникационной среды логически реализует коммуникационные уровни ВС, задает нумерацию ЭМ и распределение процессов программы по ним (cpu affinity).

Результатом работы системы оценки является время выполнения коллективной операции «all-to-all» на заданной конфигурации подсистем ЭМ. После получения оценки времени выполнения коллективной операции «all-to-all» для различных конфигураций подсистем ЭМ одного ранга устанавливается

отношение порядка выбора подсистемы ЭМ, исходя из минимума времени реализации информационных обменов В табл. 3 представлены результаты экспериментального исследования предложенного алгоритма формирования подсистем ЭМ в сравнении с временем выполнения операции «all-to-all» на ВС, состоящей из 6 NUMA-узлов, соединенных сетью связи InfiniBand (библиотека MVAPICH 2.3). Размер передаваемого сообщения равен 1 Мбайт.

Табл. 3. Время выполнения операции «all-to-all»

Ранг подсистемы	Время выполнения операции «all-to-all», с		
	1 ЭМ, 2 ядра	2 ЭМ, 1 ядро	
2			
Прогноз, с	0,00016	0,00033	
Экспериментальный запуск, с	0,00044	0,00061	
Установленный порядок	1	2	
4	1 ЭМ, 4 ядра	2 ЭМ, 2 ядра	4 ЭМ, 1 ядро
Прогноз, с	0,0019	0,0021	0,0018
Экспериментальный запуск, с	0,0031	0,0036	0,0029
Установленный порядок	2	3	1
8	1 ЭМ, 8 ядер	2 ЭМ, 4 ядер	4 ЭМ, 2 ядер
Прогноз, с	0,00384	0,19	0,0058
Экспериментальный запуск, с	0,00754	0,09	0,0076
Установленный порядок	1	3	2

Эксперименты показали хорошую согласованность результатов теоретической оценки времени выполнения операции «all-to-all» и реального времени выполнения на вычислительном кластере.

Четвертая глава посвящена развитию мультикластерной вычислительной системы, в конфигурировании которой автор принимал активное участие. Выполнено описание конфигурации системы и созданных средств организации мультипрограммного функционирования ВС.

Разработанные программные средства позволяют повысить эффективность эксплуатации ресурсов ВС при выполнении параллельных MPI-программ.

В заключении сформулированы основные результаты, полученные в диссертационной работе, а также перспективы дальнейшей разработки темы.

ЗАКЛЮЧЕНИЕ

В диссертации предложены модели, алгоритмы и программные средства организации функционирования ВС в мультипрограммном режиме обслуживания потока задач с нефиксированными параметрами и эвристический алгоритм формирования подсистем ЭМ с учетом возможной деградации производительности каналов межмашинных связей при их одновременном (конкурентном) использовании процессами параллельной программы.

1. Предложены алгоритмы построения расписания решения задач с нефиксированными параметрами.

1.1. Разработан генетический алгоритм построения расписаний решения задач с нефиксированными параметрами, обеспечивающий сокращение суммарного времени обработки набора задач в среднем на 45 % по сравнению с

детерминированными алгоритмами с гарантированной оценкой точности FFDH и VFDH.

1.2. Для обслуживания больших наборов задач с нефиксированными параметрами предложены и апробированы параллельные генетические алгоритмы, основанные на методике крупноблочного распараллеливания и характеризующиеся ускорением, близким к линейному.

1.3. Для интеграции с современными системами управления ресурсами ВС создан программный инструмент поддержки режима обслуживания потока задач с нефиксированными параметрами в системе управления ресурсами TORQUE, позволяющий сократить суммарное время решения задач набора в среднем на 24 % относительно стандартных методов.

1.4. Для планирования решения задач с нефиксированными параметрами, разработан программный инструмент поддержки режима обслуживания потока задач с нефиксированными параметрами для планировщика Maui, позволяющий сократить время обработки набора задач в среднем на 21 % по сравнению с методом «обратного заполнения» (backfilling).

2. Предложены модели и алгоритмы формирования подсистем ЭМ для выполнения параллельных программ на вычислительных системах.

2.1. Разработаны алгоритмические и программные средства оценки времени реализации коммуникационных операции типа «all-to-all», учитывающие падение производительности каналов связи иерархически организованной коммуникационной сети ВС при их конкурентном использовании процессами параллельных MPI-программ.

2.2. Создан алгоритм формирования подсистем ЭМ минимизирующий время реализации коллективных обменов типа «all-to-all» и учитывающий загруженность каналов связи, возникающую в следствии их конкурентного использования процессами параллельных программ. На вычислительных кластерах с многопроцессорными NUMA-узлами и сетью связи стандарта InfiniBand алгоритм обеспечивает сокращение времени информационных обменов от 16% до 31% по сравнению с известным алгоритмом FF (first fit).

3. Выполнено развитие программно-аппаратной конфигурации мультикластерной ВС. Программный инструмент системы расширен разработанными автором пакетами организации мультипрограммного обслуживания потоков параллельных задач с нефиксированными параметрами.

Перспективы дальнейшей разработки темы связаны с развитием нетрудоёмких методов обработки наборов задач с нефиксированными параметрами для более масштабных ВС с числом ЭМ порядка 10^7 .

ОПУБЛИКОВАННЫЕ РАБОТЫ АВТОРА ПО ТЕМЕ ДИССЕРТАЦИИ

Статьи в журналах из перечня ВАК РФ

1. Ефимов, А.В. Модернизация системы управления ресурсами PBS/TORQUE и планировщика Maui для обслуживания масштабируемых задач / А.В. Ефимов С.Н.

Мамойленко, **Е.Н. Перышкова** // Вестник компьютерных и информационных технологий. – 2016. – № 2. – С. 34-39.

2. **Перышкова, Е.Н.** Инструментарий решения масштабируемых задач на распределённых вычислительных системах / Е.Н. Перышкова, А.В. Ефимов, С.Н. Мамойленко // Вестник СибГУТИ. – 2015. – №4. – С. 82-89.

3. Ефимов, А.В. Организация функционирования распределённых вычислительных систем при обработке наборов масштабируемых задач / А.В. Ефимов, С.Н. Мамойленко, **Е. Н.Перышкова** // Вестник ТГУ. Управление, вычислительная техника и информатика. – 2011. – № 2(15). – С. 51-60.

4. Хорошевский, В.Г. Масштабируемый инструментарий параллельного мультипрограммирования пространственно-распределённых вычислительных систем / В.Г. Хорошевский, М.Г. Курносов, С.Н. Мамойленко, К.В. Павский, А.В. Ефимов, А.А. Пазников, **Е.Н. Перышкова** // Вестник СибГУТИ. – 2011. – № 4. – С. 3-18.

Публикации в изданиях, индексируемых Scopus

5. **Peryshkova, E.** Experimental Study of Network Contention Effects on All-to-All Operation / E. Peryshkova, M. Kurnosov // Proc. of the 14th International Scientific-Technical Conference «Actual Problems of Electronic Instrument Engineering» (APEIE-2018), 2018. – Vol. 1, Part 4. – P. 506-510.

Свидетельства о государственной регистрации программ для ЭВМ

6. Свидетельство 2018619237 РФ. Программа моделирования алгоритмов коммуникационных обменов All-to-all на вычислительных кластерах: свидетельство о государственной регистрации программы для ЭВМ / **Е.Н. Перышкова**, М.Г. Курносов; заявитель и патентообладатель СибГУТИ; опубл. 02.08.2018.

7. Свидетельство 2016618446 РФ. Модуль планировщика MAUI, организующий работу масштабируемых задач на распределённых вычислительных системах: свидетельство о государственной регистрации программы для ЭВМ / **Е.Н. Перышкова**, А.В. Ефимов, С.Н. Мамойленко; заявитель и патентообладатель ИФП СО РАН; опубл. 20.08.2016.

8. Свидетельство 2016618768 РФ. Модуль системы управления ресурсами TORQUE организующий работу масштабируемых задач на распределённых вычислительных системах: свидетельство о государственной регистрации программы для ЭВМ / **Е.Н. Перышкова**, А.В. Ефимов, С.Н. Мамойленко; заявитель и патентообладатель СибГУТИ; опубл. 05.08.2016.

9. Свидетельство 2012613762 РФ. Программа моделирования и анализа алгоритмов формирования расписаний решения масштабируемых задач на распределённых вычислительных системах: свидетельство о государственной регистрации программы для ЭВМ / В.Г. Хорошевский, А.В. Ефимов, С.Н. Мамойленко, **Е.Н. Перышкова**, К.Е. Крамаренко, Р.И. Кожушко; заявитель и патентообладатель СибГУТИ; опубл. 20.04.2016.

Публикации в сборниках трудов и материалах конференций

10. **Перышкова, Е.Н.** Моделирование конкурентного разделения каналов связи при реализации операции All-To-All / Е.Н. Перышкова, М.Г. Курносов // Труды XIV Международной научно-технической конференции «Актуальные проблемы электронного приборостроения». – Новосибирск, 2018. – Т. 6. – С. 151-155.

11. **Перышкова, Е.Н.** Формирование подсистем элементарных машин в вычислительных системах с учетом конкурентного разделения каналов связи / Е.Н. Перышкова, М.Г. Курносов // Материалы XII Российской конференции с международным участием «Новые информационные технологии в исследовании сложных структур». – Томск, 2018. – С. 8-9.

12. **Перышкова, Е.Н.** Влияние размещения процессов параллельной программы по вычислительным узлам на эффективность операции All-to-all / Е.Н. Перышкова, М.Г. Курносов // Материалы Российской научно-технической конференции «Обработка информации и математическое моделирование», 2018. – С. 299-304.

13. **Перышкова, Е.Н.** Формирование подсистем элементарных машин в вычислительных кластерах на базе составных коммутаторов // Сборник трудов тринадцатой международной Азиатской школы-семинара «Проблемы оптимизации сложных систем» в рамках международной конференции IEEE SIBIRCON-2017. – Новосибирск, 2017. – С. 85-89.

14. **Перышкова, Е.Н.** Моделирование конкурентного разделения каналов связи при реализации параллельных программ на мультиархитектурных вычислительных системах / Е.Н. Перышкова, М.Г. Курносов // Сборник докладов международной конференции «Визуальная аналитика». – Кемерово, 2017. – С. 62-65.

15. **Перышкова, Е.Н.** Анализ влияния конкурентного использования коммуникационной среды вычислительных систем на динамические характеристики параллельных программ // Сборник статей всероссийской научно-практической конференции с международным участием «Интеллектуальный анализ сигналов, данных и знаний: методы и средства». – Новосибирск, 2017. – С. 206-210.

16. **Перышкова, Е.Н.** Анализ масштабируемости алгоритмов обслуживания потоков задач в системе управления ресурсами SLURM // Материалы российской научно-технической конференции «Обработка информации и математическое моделирование». – Новосибирск, 2017. – С. 357-363.

17. Ефимов, А.В. Исследование эффективности обработки масштабируемых задач на распределенных вычислительных системах / А.В. Ефимов, С.Н. Мамоиленко, **Е.Н. Перышкова** // Сборник докладов национального суперкомпьютерного форума. – Переславль-Залесский, 2016. – 9 с.

18. **Перышкова, Е.Н.** Использование масштабируемых задач на распределенных вычислительных системах / Е.Н. Перышкова, А.В. Ефимов, С.Н. Мамоиленко // Материалы Российской научно-технической конференции «Перспективные информационные и телекоммуникационные технологии», 2016. – С. 234-236.

19. Ефимов, А.В. Программное обеспечение на основе PBS/Torque и планировщика MAUI для обслуживания масштабируемых задач / А.В. Ефимов, С.Н. Мамоиленко, **Е.Н. Перышкова** // Доклады второй международной конференции «Инжиниринг & Телекоммуникации». – Москва, 2015. – С. 188-190.

20. **Перышкова, Е.Н.** Инструментарий решения масштабируемых задач на распределенных вычислительных системах / Е.Н. Перышкова, А.В. Ефимов, С.Н. Мамоиленко // Сборник трудов восьмой Сибирской конференция по параллельным и высокопроизводительным вычислениям. – Томск, 2015. – С. 143-150.

21. **Перышкова, Е.Н.** Планирование решения масштабируемых задач на ресурсах распределенных вычислительных систем / Е.Н. Перышкова, А.В. Ефимов, С.Н. Мамоиленко // Тезисы международной конференции «Актуальные проблемы вычислительной и прикладной математики» посвященной 90-летию со дня рождения академика Марчука. – Новосибирск, 2015. – С. 110.

Автореферат

Перышкова Евгения Николаевна

**СРЕДСТВА УПРАВЛЕНИЯ РЕСУРСАМИ ВЫЧИСЛИТЕЛЬНЫХ
СИСТЕМ В РЕЖИМЕ ОБСЛУЖИВАНИЯ ПОТОКА ЗАДАЧ
С НЕФИКСИРОВАННЫМИ ПАРАМЕТРАМИ**

Автореферат диссертации на соискание ученой степени
кандидата технических наук

Подписано в печать «24» октября 2018 г.

Бумага офсетная. Печать цифровая.

Усл. печ. л. 1,5.

Тираж 150. Заказ 1024.

Отпечатано в ООО «Параллель»

630090, Новосибирск, ул. Институтская, 4/1. Тел. (383) 330-26-98.