

На правах рукописи

Теплов Алексей Михайлович

**АНАЛИЗ МАСШТАБИРУЕМОСТИ ПАРАЛЛЕЛЬНЫХ  
ПРИЛОЖЕНИЙ НА ОСНОВЕ ТЕХНОЛОГИЙ  
СУПЕРКОМПЬЮТЕРНОГО КОДИЗАЙНА**

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

Автореферат  
диссертации на соискание учёной степени  
кандидата физико-математических наук

Москва – 2015

Работа выполнена в лаборатории параллельных информационных технологий Научно-исследовательского вычислительного центра Московского государственного университета имени М.В.Ломоносова".

**Научный руководитель:** доктор физико-математических наук, профессор, член-корреспондент РАН, Воеводин Владимир Валентинович

**Официальные оппоненты:** Кореньков Владимир Васильевич доктор технических наук, Объединенный институт ядерных исследований (ОИЯИ), директор лаборатории информационных технологий Суков Сергей Александрович кандидат физико-математических наук, Институт прикладной математики им. М.В. Келдыша Российской академии наук, старший научный сотрудник

**Ведущая организация:**

Южно-Уральский государственный университет (национальный исследовательский университет) ФГБОУ ВПО «ЮУрГУ» (НИУ)

Защита состоится 13 ноября 2015 года в 15 часов на заседании диссертационного совета Д 501.002.09 Московского государственного университета имени М.В. Ломоносова по адресу: 119991, г. Москва, Ленинские горы, д.1, стр. 4, НИВЦ МГУ, конференц-зал.

С диссертацией можно ознакомиться в Научной библиотеке МГУ имени М.В. Ломоносова (Ломоносовский проспект, 27) и на сайте: <http://srcc.msu.ru/nivc/sci/dissert/dissert.html>

Автореферат разослан «\_\_» \_\_\_\_\_ 2015 года.

Учёный секретарь  
диссертационного совета

Суворов Владимир Викторович

## Общая характеристика работы

### Актуальность темы

Современные научные и индустриальные приложения и пакеты прикладных программ используют очень большие объемы данных, для обработки которых зачастую необходимы колоссальные объемы вычислений. Решать эти задачи часто приходится в близком к реальному времени темпе, что дополнительно повышает требования к производительности вычислительных систем. Это требует применения вычислительных устройств с очень высокой производительностью. Сейчас для этих целей, как правило, применяют многопроцессорные вычислительные системы, выполняющие большой набор различных параллельных программ. Вместе с этим, эффективность выполнения этих программ на существующих вычислительных системах, как правило, крайне низка. Чаще всего это связано со сложностью создания программ, оптимально сочетающих алгоритм решаемой задачи и его реализацию в программном коде. Проблема заключается в необходимости эффективного использования параллельной программой десятков миллионов компонентов больших вычислительных систем, которые или имеются уже сейчас, или неизбежно появятся в ближайшем будущем.

Суперкомпьютерный кодизайн – это процесс разработки параллельных приложений, в котором учитываются и согласуются особенности алгоритмов, технологий программирования, реализации в коде и аппаратном обеспечении для создания высокоэффективных программ для вычислительных систем. Современные концепции суперкомпьютерного кодизайна требуют, чтобы особенности функционирования каждой составляющей вычислительной системы принимались во внимание при разработке параллельной программы. Поэтому при создании параллельных программ требуется анализ и учет большого объема информации о свойствах каждой составляющей вычислительного процесса. Требования к параллелизму работы должны быть учтены на всех этапах решения задачи: в алгоритмах, в программах, в архитектуре компьютеров и – ключевой вопрос — в согласованности между собой найденных особенностей на каждом из этапов во время выполнения программы.

Вопрос эффективного использования вычислительных систем, способных справиться с большими и сложными расчетами, напрямую связан с понятием «масштабируемость». Масштабируемость – свойство параллельной программы, характеризующее зависимость её динамических характеристик от параметров запуска этой программы. Анализ масштабируемости программы позволяет делать выводы о качестве ее работы. Нельзя оптимизировать параллельную программу без учета параметров запуска программы и других факторов, влияющих на её динамические характеристики. Особенно актуаль-

но это при разработке сложных программных комплексов, в которых изменения кода в процессе оптимизации в одной части программы вполне могут снижать масштабируемость других ее частей и всего программного комплекса в целом.

При разработке больших программных комплексов часто возникает вопрос правильного выбора алгоритма и его реализации. Для выбора эффективного варианта реализации разрабатываемой программы необходимо иметь простую и надежную методику сравнения свойств масштабируемости различных реализаций программы. Это может дать возможность целенаправленно выбирать наиболее удачные реализации использованного алгоритма или даже использовать алгоритмы другого класса и, при необходимости, скорректировать конфигурацию аппаратной платформы. Следовательно, для решения этих задач необходимо иметь конструктивный подход для учета всех факторов, влияющих на масштабируемость программы.

Сегодня параллельные вычисления встречаются повсеместно: многоядерные процессоры используются даже в наручных часах, а восьмиядерный процессор в мобильном телефоне становится стандартным. Степень параллелизма в вычислительных системах высокого уровня достигает нескольких миллионов, и создание эффективных программ требует тщательного анализа их свойств. Все это в совокупности говорит о том, что вопрос разработки методов исследования масштабируемости параллельных программ на сегодня является крайне актуальным.

### **Степень разработки темы**

Вопросу исследования масштабируемости как небольших программ, так и больших программных комплексов в настоящее время посвящено немало исследований, результаты которых приведены во множестве научных публикаций. В этих работах даны различные определения этого свойства, причем большое разнообразие имеющихся в литературе определений масштабируемости говорит о необходимости дальнейшего поиска более простого и универсального определения, приемлемого для широкого круга пользователей и разработчиков параллельных программ.

Каждый из использованных авторами этих публикаций подходов к анализу масштабируемости специфичен для своей области применения, хотя они и содержат общие элементы. Одни методы просты в применении, но дают существенные погрешности результатов. Другие методы дают приемлемые результаты, но очень сложны и, следовательно, достаточно дороги. И часто они все равно не позволяют сделать однозначных выводов о причинах найденных особенностей масштабируемости. Технологии суперкомпьютерного кодизайна предполагают проведение наиболее полного анализа особенностей работы программ на всех этапах разработки для согласованной работы программ и аппаратного обеспечения. Но такой анализ обычно не направ-

лен на исследование масштабируемости программ. Поэтому, несмотря на значительное многообразие существующих методов исследования масштабируемости, этот вопрос все еще требует дальнейшего и всестороннего развития. Не менее важно, чтобы методы анализа были универсальны и применимы для анализа большинства параллельных программ. Желательно, чтобы исследование по подобной методике могло быть проведено по принципу «черного ящика», а его проведение могло бы быть на каких-то стадиях автоматизировано.

Многие российские и зарубежные исследователи внесли серьезный вклад в исследование понятия масштабируемости, среди них: В.В.Воеводин, В.А.Крюков, И.Б.Задыхайло, В.П.Ильин, В.Э.Малышкин, М.В.Якововский, J.Dongarra, T.Sterling, V.Gropp, T.Hoefler и другие.

### **Цель работы и вытекающие из нее задачи**

Целью данной работы является развитие методов исследования свойства масштабируемости параллельных программ, разработка подходов для определения причин и степени влияния на масштабируемость программных и аппаратных факторов с использованием технологий суперкомпьютерного кодизайна, а также разработка методики сравнения масштабируемости различных параллельных программ для их последующей оптимизации.

Для достижения указанной цели в процессе работы решались следующие задачи:

- разработка подхода к анализу масштабируемости параллельных приложений на основе технологий суперкомпьютерного кодизайна для анализа свойств и характеристик выполнения приложений;
- разработка подхода к анализу причин, ограничивающих масштабируемость параллельных приложений, на основе данных системного мониторинга как источника данных о состоянии аппаратного обеспечения во время работы программы;
- разработка метода сравнения масштабируемости параллельных приложений для получения критериев оценки вносимых изменений в приложение или аппаратную платформу, а также для сравнения масштабируемости различных программ;
- проведение апробации предложенных в работе подходов и методов в рамках анализа тестовых параллельных программ, параллельных реализаций типовых алгоритмов и пользовательских программ из потока задач суперкомпьютерного комплекса Московского университета.

### **Научная новизна**

Научная новизна предлагаемой работы заключается в разработке и обосновании определения масштабируемости параллельной программы, обобщающего существующие определения этого свойства. Разработаны методики сбора и анализа данных о масштабируемости параллельных программ. Вве-

дены понятия параметров запуска параллельной программы, конфигурации запуска, динамических характеристик, предложена классификация данных понятий. С использованием технологий суперкомпьютерного кодизайна разработана методика анализа данных системного мониторинга, позволяющая не только выявлять проблемы масштабируемости параллельных программы, но и находить причины этих проблем путем сопоставления информации о различных характеристиках программ и сбора данных с различных источников программно-аппаратной среды суперкомпьютера. В работе предложены подход к сравнению масштабируемости различных параллельных программ и методика расчета метрики масштабируемости, основанная на собранных данных о масштабируемости программы.

### **Новые научные результаты и положения, выдвигаемые для публичной защиты**

Основными результатами работы являются:

1. Предложен и обоснован подход к анализу масштабируемости параллельных приложений на основе технологий суперкомпьютерного кодизайна.

2. Предложен подход к выделению истинных причин, ограничивающих масштабируемость параллельных приложений, на основе согласованного анализа данных системного мониторинга.

3. Предложен метод сравнения масштабируемости параллельных приложений, основанный на анализе изменения эффективности их реализаций. Метод позволяет не только оценивать эффект от внесенных в программу изменений, но и сравнивать масштабируемость любых параллельных программ между собой.

4. Предложенные в работе подходы и методы прошли успешную апробацию в рамках анализа тестовых параллельных программ, параллельных реализаций типовых алгоритмов и пользовательских программ из потока задач суперкомпьютерного комплекса Московского университета.

### **Теоретическая и практическая значимость работы**

Разработанный подход к исследованию масштабируемости позволяет определять свойства масштабируемости параллельных программ по многим параметрам запуска, а также сочетать в единой структуре данные о влиянии различных параметров запуска программ на их выполнение. Разработанный подход представляет теоретическую ценность как расширение понятия масштабируемости параллельных программ, позволяющее конструктивно подойти к исследованию их свойств. Предлагаемый подход позволяет делать выводы о причинах найденных проблем масштабируемости параллельных программ, проводить сравнение программ по данному свойству. С использованием предложенного подхода к анализу масштабируемости в работе предложен метод сравнения масштабируемости различных программ, что пред-

ставляет практическую ценность при разработке параллельных масштабируемых приложений.

#### **Методология и методы исследования**

Для выполнения поставленной цели в данной работе применялся подход, основанный на:

- методах анализа структуры алгоритма решаемых задач;
- методах синтаксического анализа исходного кода программ;
- анализе динамических характеристик программ;
- анализе данных реальных трасс выполнения параллельных программ;
- сопоставлении найденных особенностей программы на разных этапах решения задачи с применением технологий кодизайна.

#### **Степень достоверности и апробация работы**

Предложенные в работе подходы и методики прошли апробацию и обсуждение:

- на научных семинарах НИВЦ МГУ и кафедры СКИ факультета ВМК МГУ;
- на спецсеминаре «Вопросы распределенной обработки информации» факультета ВМК МГУ;
- на Научной конференции «Ломоносовские чтения» (г. Москва, 2012, 2013, 2014 гг.);
- на Всероссийской научной конференции «Научный сервис в сети Интернет: поиск новых решений» (г. Новороссийск, 2012г.);
- на Всероссийской научной конференции «Научный сервис в сети Интернет: все грани параллелизма» (г. Новороссийск, 2013г.);
- на Всероссийской научной конференции «Научный сервис в сети Интернет: многообразие суперкомпьютерных миров» (г. Новороссийск, 2014г.);
- на XIV Международной конференции «Высокопроизводительные параллельные вычисления на кластерных системах», ПНИПУ, (г. Пермь, 2014г.);
- на IX Международной конференции «Параллельные вычислительные технологии (ПаВТ'2015)», (г. Екатеринбург 2015)
- на научном семинаре «Методы суперкомпьютерного моделирования» (г. Таруса, 2015г.)

**Личный вклад автора.** Все исследования, результаты которых изложены в диссертационной работе, проведены лично автором в процессе научной деятельности. Из совместных публикаций в результаты диссертационной работы включен лишь тот материал, который непосредственно принадлежит автору.

**Структура и объем работы.** Диссертация состоит из введения, 4 глав, заключения и списка литературы. Общий объем диссертации – 141 страница. Список литературы включает 54 наименования.

## 2 Основное содержание работы

Во **введении** диссертационной работы обоснована необходимость расширенного исследования масштабируемости параллельных программ. Указаны актуальность темы и степень ее разработки, сформулированы цель исследования и вытекающие из нее задачи.

В **первой главе** работы дан краткий обзор основных методов и подходов, применяемых при исследовании выполнения параллельных программ на больших вычислительных системах.

В опубликованных научных работах приведено большое разнообразие методов, применяемых при анализе выполнения параллельных программ. Все эти методы можно условно разделить на три основные группы: аналитические, статический анализ исходного текста и эмпирические. Применяются и смешанные методы, сочетающие черты основных групп.

Выполнение параллельной программы может характеризоваться разными способами, и при этом применяются различные показатели хода выполнения программы. Существует большой набор показателей выполнения программы. Влияние на выполнение программы может быть охарактеризовано, начиная с влияния особенностей алгоритмической основы, особенностей математического метода, его программной реализации и аппаратных и заканчивая особенностями выбранной аппаратной платформы. Также организация выполнения параллельной программы подразумевает наличие накладных расходов на распараллеливание и расходов, напрямую не связанных с выполняемым алгоритмом.

Процесс разработки высокоэффективных параллельных программ, учитывающий особенности на всех этапах разработки и выполнения программ и позволяющий согласовать их для достижения наибольшей эффективности выполнения, называется **суперкомпьютерным кодизайном**. Технологии суперкомпьютерного кодизайна позволяют проводить анализ влияния таких особенностей на выполнение программы. Для анализа масштабируемости особый интерес представляет изменение таких особенностей выполнения при изменении параметров запуска приложения. Потому использование технологий суперкомпьютерного кодизайна для анализа масштабируемости является важным источником данных о причинах проблем масштабируемости программ.

Большое число показателей выполнения программы позволяет с большей точностью анализировать ее работу. Одновременное рассмотрение всех показателей позволяет более детально отражать особенности выполнения параллельной программы на вычислительной системе.



Существующие подходы к анализу динамических характеристик параллельных программ можно разделить на несколько основных частично пересекающихся групп.

*Профилирование* – сбор статистической информации о времени выполнения различных частей программы и частоте их вызова.

*Трассировка* – сбор информации о порядке выполнения в исследуемой программе определенных заранее событий.

*Моделирование работы* – создание модели исследуемого приложения и вычисление на ее основе характеристик работы программы.

*Эмуляция вычислений* – выполнение реальных вычислений меньшего масштаба, с экстраполяцией на интересующую конфигурацию системы.

Существуют также методики, комбинирующие и дополняющие эти подходы.

Подходы к анализу работы параллельных программ различаются не только по способу получения данных об их работе, но и по характеру получаемых данных.

В главе проведен обзор определений понятия масштабируемость из большого числа источников. Показано разнообразие определений, которое свидетельствует о том, что это понятие еще не стало устоявшимся и потому при исследовании его каждый автор использует наиболее интересные для него показатели работы программы и изменяемые параметры.

Подходы к исследованию масштабируемости очень часто совпадают с основными методами исследования динамических характеристик параллельных программ. Процесс исследования масштабируемости параллельных программ предполагает получение необходимых данных о динамических характеристиках для всей области изменений параметров запуска. Дальнейшее исследование чаще всего сводится к анализу изменений и поиску причин, оказавших влияние на рассмотренные характеристики.

Для таких подходов можно использовать уже существующий инструментарий анализа параллельных программ. Однако они обладают недостаточной информативностью и потому поиск истинных причин изменений характеристик является достаточно сложным.

**Вторая глава** работы посвящена теоретическому анализу понятия «масштабируемость» параллельных программ, и в ней дан пример апробации предлагаемого подхода к анализу масштабируемости параллельного приложения.

Масштабируемость является одним из важнейших свойств параллельной программы. В научных публикациях можно найти достаточно большое число различных определений этого свойства. Рассмотрены и детально разобраны несколько определений, имеющихся в литературе. Показаны различия в понимании этого свойства и общность в понимании процесса исследования.

Основным недостатком большинства имеющихся определений масштабируемости является то, что это понятие рассматривается авторами чаще всего при изменении какого-то одного параметра запуска. При проведении исследований такой подход приводит к большому разбросу результатов, потому что параметров, оказывающих влияние на выполнение программы, значительно больше. Рассмотрение какого-то одного из показателей выполнения программы не отражает полной картины свойства масштабируемости параллельной программы. Другим недостатком является то, что авторы рассматривают разные параметры и разные показатели, что ведет к невозможности сопоставления полученных результатов исследований разных авторов.

Масштабируемость требует значительно более глубокого анализа и рассмотрения при этом большего и более разнообразного вида динамических характеристик программы, чем в формулировках, имеющихся в научных публикациях. Особенно имея возможность анализировать особенности приложения с помощью технологий суперкомпьютерного кодизайна.

На основании приведенных доводов определены понятия, используемые далее в исследовании, и введено определение масштабируемости, позволяющее обобщить существующие исследования без указанных противоречий.

**Динамическими характеристиками** параллельной программы будем называть любые показатели выполнения параллельной программы. Это могут быть или средние характеристики за период времени или моментальные характеристики, актуальные в какой-то момент работы. Динамические характеристики можно разделить по способу получения значений на две группы.

К первой группе можно отнести **измеряемые характеристики**. Такими характеристиками могут быть время выполнения, количество кэш-промахов, количество переданных за время работы байт по сети или по шине данных, количество обращений в память за время работы и т.п.

Во вторую группу включены **вычисляемые характеристики**, значения которых можно получить путем вычислений из измеряемых и других вычисляемых характеристик. Такие характеристики используются для лучшего понимания работы программы и более наглядного представления результатов. Например, это ускорение, производительность, скорость передачи данных, число кэш-промахов в секунду, моментальная и средняя эффективность и т.п.

**Параметрами запуска** параллельной программы будем называть любые параметры, изменяемые при запуске, оказывающие влияние на выполнение параллельной программы (например: размерность матрицы с данными, число процессов, размер блока матрицы, число процессов на узел и т.п.).

**Конфигурация запуска** – совокупность параметров запуска, при которых осуществляется запуск параллельной программы при каждом конкретном ее исследовании.

Для анализа особенностей масштабируемости параллельной программы и причин этих особенностей важно рассматривать изменения всех доступных динамических характеристик при всех параметрах запуска. Для этого может применяться весь спектр доступных технологий суперкомпьютерного кодирования для анализа свойств и изменений динамических характеристик при выполнении алгоритмов, так и согласованности с особенностями аппаратной платформы.

Поэтому **масштабируемостью** целесообразно называть **свойство параллельной программы, характеризующее зависимость изменения всей совокупности динамических характеристик работы этой программы от множества параметров ее запуска.**

Использование при анализе свойств параллельной программы термина «масштабируемость» в такой формулировке позволяет наиболее полно отражать особенности параллельной программы и получать представление о том, как меняется использование вычислительных ресурсов при изменении параметров запуска программы. И уже на основании этого делать выводы о факторах, влияющих на работу программы.

Рассмотрение большого числа динамических характеристик при изменении нескольких параметров запуска требует наличия значительного объема данных о выполнении программы и, следовательно, проведения большого числа экспериментов. Это определяется необходимостью получения данных о работе исследуемой программы на каждой конфигурации в рассматриваемой области значений параметров запуска.

Таким образом, предлагаемые принципы исследования масштабируемости подразумевают объединение всех доступных данных о работе параллельной программы и анализ влияния как каждого отдельного фактора, так и их совместного воздействия на процесс выполнения параллельной программы. При этом данные о масштабируемости программы можно представить в виде векторной функции от нескольких переменных. Аргументами такой функции будут являться значения параметров запуска, оказывающих влияние на ход выполнения программы. А значения функции можно представить в виде вектора значений динамических характеристик, полученных при выполнении параллельной программы. К этим динамическим характеристикам нужно отнести как вычисляемые динамические характеристики (время выполнения, ускорение, эффективность, производительность), так и динамические характеристики, полученные в результате работы системы мониторинга (кэш-промахи, загрузка сетей передачи данных, интенсивность использования своп-памяти и т.д.) Отсюда следует очень важное свойство понятия масштабируемости – **многомерность**. Масштабируемость является многомерной: каждый параметр запуска – измерение, а каждое выполнение характеризуется большим вектором показателей.

В главе показана непротиворечивость введенного многомерного понятия существующим определениям, которые рассматриваются в виде частных случаев многомерного случая с фиксированным набором параметров запуска.

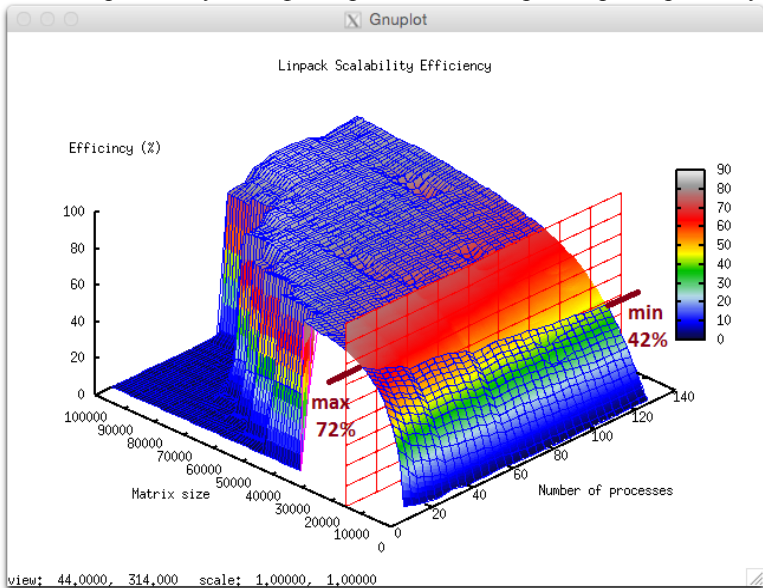


Рисунок 1. Масштабируемость теста Linpack benchmark. График эффективности выполнения программы. Плоскостью дано сечение графика при размере матрицы в 15000. В сечении обозначены максимум в 72% и минимум в 42% значений эффективности

Предлагаемый подход к понятию масштабируемости позволяет учитывать влияние изменения различных факторов, воздействующих на масштабируемость. В результате можно находить значения параметров запуска, при которых работа программы имеет наиболее высокую производительность и достаточную эффективность, и делать выводы о том, какие из факторов и свойств вычислительной системы больше других влияют на ход выполнения параллельной программы. В свою очередь, информация о наличии проблем и их причинах позволяет более обоснованно оптимизировать конкретную параллельную программу.

В качестве примера предлагаемого подхода к анализу масштабируемости параллельной программы **во второй главе** приведен вариант исследования и последующего анализа реализации теста Linpack benchmark.

При исследовании указанного теста применялась система мониторинга, установленная на суперкомпьютере «Ломоносов», данные с которой анализировались с помощью системы Job Digest и инструментария WEKA. При этом анализировались как численные показатели датчиков, так и динамика изменения их значений. При исследовании изменялись два параметра запуска: число процессов – [8, 128] с шагом в 8 процессов и размер матрицы – [1000, 100 000] с шагом в 1000. Другие параметры запуска были постоянны.

Полученные результаты представлены на рисунке 1 в виде поверхности, для которой координаты X и Y соответствуют числу процессов и размеру матрицы соответственно, а координата Z – эффективности работы теста.

**Третья и четвертая** главы работы посвящены анализу факторов оказывающих влияние на масштабируемость программ, разработке подхода к оценке изменений, вносимых в программу для уменьшения такого влияния и их практическому применению к исследованию масштабируемости параллельных программ.

В **третьей главе** предложена и обоснована на практических примерах методика применения данных системного мониторинга для исследований масштабируемости параллельных программ. В результате проведенных экспериментов было определено, что информация, получаемая от системы мониторинга, позволяет делать выводы не только о *наличии* влияния определенных факторов, но и о *степени* их влияния на работу параллельной программы.

Анализ данных системного мониторинга работы вычислительной системы является одним из наиболее перспективных направлений анализа выполнения параллельных программ с помощью технологий суперкомпьютерного кодизайна. Прежде всего потому, что на основе этих данных можно сделать выводы о реальной производительности всей вычислительной системы и влиянии особенностей алгоритмов и их реализации в программном коде на работу аппаратного обеспечения. При разработке приложений в концепции суперкомпьютерного кодизайна данные системного мониторинга позволяют оценить согласованность программных и аппаратных особенностей для повышения эффективности их работы.

Результаты анализа информации от системного мониторинга, собранные при работе параллельной программы, позволяют выявить и проанализировать факторы, влияющие на масштабируемость этой программы. При анализе масштабируемости важно не то, как сильно влияет каждый фактор в отдельности на масштабируемость, а как это влияние меняется при изменении параметров запуска программы. В качестве примеров анализа данных системного мониторинга для выявления факторов, влияющих на масштабируемость программ, применялись как реальные программы пользователей суперком-

пьютерных комплексов, работающие в общей очереди задач, так и специально написанные для этого тесты.

В качестве инструментальных средств для исследований работы программ на суперкомпьютере «Ломоносов» были использованы система построения отчетов о работе приложений JobDigest и инструментарий для анализа данных WEKA.

Для каждого из выявленных в процессе исследований факторов приведены примеры графиков, полученных с помощью системы JobDigest, иллюстрирующих то, каким образом рассматриваемый фактор может быть обнаружен в данных мониторинга. На основе полученных результатов была разработана методика выявления основных факторов, влияющих на масштабируемость в данных системного мониторинга.

Результаты проведенных экспериментов показали, что при использовании данных системного мониторинга могут быть выявлены факторы, оказывающие влияние на масштабируемость параллельных программ:

1) Для коммуникационной сети:

- латентность коммуникационной сети;
- пропускная способность коммуникационной сети;
- топология коммуникационной сети.

2) Для компонентов вычислительного узла компьютера:

- влияние подключения жёсткого диска;
- характеристики оперативной памяти;
- объём и характеристики кэш-памяти.

3) Для факторов, связанных с характеристиками примененного алгоритма или исследуемой программы:

- дисбаланс вычислительной нагрузки;
- предел декомпозиции данных.

В **четвертой главе** рассмотрены дальнейшие вопросы анализа масштабируемости параллельных программ, и на их основе разработаны и изложены принципы сравнения масштабируемости различных параллельных программ.

При анализе масштабируемости нескольких различных по структуре параллельных программ часто встает вопрос поиска критерия, по которому можно было бы сравнивать масштабируемость этих программ. Необходима метрика, которая позволяла бы характеризовать масштабируемость исследуемой программы так, чтобы она не зависела от особенностей программы или от особенностей ее запуска и позволяла сравнивать эту программу с другими, границы значений параметров запуска которых могли бы и не пересекаться или даже не иметь общих значений.

В качестве критерия был предложен анализ изменения эффективности реализации параллельной программы как отношения реально полученной производительности при выполнении программы к пиковой производитель-

ности системы. Тогда, разбивая всю область значений параметров запуска на  $N$  элементов, с параметрами запуска, отличающимися на один шаг, можно анализировать для каждого элемента изменение эффективности для каждого отдельного параметра и для изменения всех параметров.

Для каждого параметра  $P_i$  из набора  $M$  параметров  $P_1 \dots P_M$  изменением эффективности будет

$$\Delta E_{P_i} = \frac{\sum_{j_1, j_2, \dots, j(i-1), j(j+1), \dots, j, M=1}^2 (E(\{P_1\}_{j_1}, \{P_2\}_{j_2}, \dots, \{P_i\}_{j_i}, \dots, \{P_M\}_{j_M}) - E(\{P_1\}_{j_1}, \{P_2\}_{j_2}, \dots, \{P_1\}_{j_1}, \dots, \{P_M\}_{j_M}))}{2^{(M-1)}} \quad (1)$$

Тогда вкладом элемента по параметру  $P_i$  для случая  $M$  параметров будем называть величину

$$Est1P_i = \frac{\Delta E_{P_i} * (\{P_i\}_2 - \{P_i\}_1)}{(\max_j \{P_i\}_j - \min_j \{P_i\}_j)} \quad (2)$$

Приращением эффективности для  $M$  параметров будем называть среднее значение приращения по каждому параметру

$$\Delta E_A = \frac{\sum_{i=1}^M \Delta E_{P_i}}{M} \quad (3)$$

Тогда вкладом элемента для случая  $M$  параметров будем называть

$$Est1A = \frac{\Delta E_A * \prod_{j=1}^M (\{P_j\}_{i+1} - \{P_j\}_i)}{\prod_{j=1}^M (\max_i \{P_j\}_i - \min_i \{P_j\}_i)} \quad (4)$$

где  $\{P_i\}_j$  –  $j$ -е значение параметра запуска  $P_i$ ,  $\max\{P_i\}$ ,  $\min\{P_i\}$  – максимальное и минимальное значение параметров запуска  $P_i$ .

*Общей оценкой масштабируемости по произвольному параметру  $P_i$*  будем называть среднее значение вклада каждого элемента рассматриваемой области значений в общую оценку по рассматриваемому параметру по всем элементам рассматриваемой области параметров запуска:

$$EstP_i = \frac{\sum_{j=1}^N \{Est1P_i\}_j}{N} \quad (5)$$

*Общей оценкой масштабируемости* будем называть средний размер вклада каждого элемента рассматриваемой области значений в общую оценку по всем элементам рассматриваемой области параметров запуска:

$$EstAll = \frac{\sum_{j=1}^N Est1A_j}{N} \quad (6)$$

Тогда масштабируемость параллельного приложения на рассматриваемой области значений параметров запуска будет характеризовать *метрика масштабируемости*:

$$(\min_i\{P_i\}_j, \dots, \min_i\{PM\}_i; \max_i\{P_i\}_j, \dots, \max_i\{PM\}_i; EstP1, \dots, EstPM, EstAll; \max\{E\}, \min\{E\})$$

(7)

Где  $\max\{P_i\}_j, \min\{P_i\}_j$  – границы изменения параметра  $P_i$ , а  $\max\{E\}$  и  $\min\{E\}$  – границы изменения значений эффективности работы приложения.

В качестве примеров применения предлагаемых принципов сравнения масштабируемости приведены результаты анализа:

- программы, реализующей блочное перемножение матриц;
- программы, реализующей скалярное произведение векторов;
- программы, реализующей разложение Холецкого;
- программы, реализующей тест Linpack Banchmark;
- программы, реализующей прямой ход метода Гаусса решения СЛАУ.

Проанализированы результаты сравнения масштабируемости исследованных программ. Кроме указанных программ, в процессе работы над диссертацией исследовались также и ряд других программ: метод сдвигания, параллельное суммирование элементов вектора, умножение матрицы на вектор и другие. Все рассмотренные программы исследовались при разработке подхода к сравнению масштабируемости различных параллельных программ в рамках проекта создания Открытой энциклопедии свойств алгоритмов AlgoWiki.

### Заключение

Предложенный подход к пониманию свойства масштабируемости параллельных программ имеет высокую значимость и отражает сложность и многогранность этого свойства программ. Актуальность задачи разработки подхода к анализу масштабируемости является высокой ввиду необходимости описания свойств параллельных программ и поиска истинных причин плохой масштабируемости параллельных приложений.

Предложенный подход к исследованию масштабируемости параллельных приложений, использующий технологии суперкомпьютерного кодизайна для анализа всех уровней выполнения программы, позволяет описывать свойства параллельных программ и находить истинные факторы, снижающие масштабируемость программ.

Предложенный подход к сравнению масштабируемости приложений на основе анализа изменения эффективности их реализаций позволяет оценивать эффект от внесенных изменений на любом уровне выполнения прило-



жения. Также подход позволяет сравнивать масштабируемость любых параллельных программ между собой.

Исходя из полученных результатов в ходе апробации, можно рекомендовать разработанные в рамках предложенного подхода методы анализа для использования при проведении исследования истинных причин снижения масштабируемости, разработке программ с использованием технологий кода-дизайна и оптимизации выполнения существующих программ на параллельных вычислительных системах.

#### **4 Список работ, в которых опубликованы основные положения диссертации**

##### **В журналах из Перечня ВАК:**

1. Антонов, А.С., Жуматий, С.А., Никитенко, Д.А., Стефанов, К.С., Теплов, А.М., Швец, П.А. Исследование динамических характеристик потока задач суперкомпьютерной системы //Вычислительные методы и программирование: Новые вычислительные технологии (Электронный научный журнал). – М.: Изд-во МГУ, 2013. – Том 14 – С. 104-108 (<http://num-meth.srcc.msu.ru/>).

2. Антонов, А. С., Теплов, А.М. Использование данных системного мониторинга для определения факторов, уменьшающих масштабируемость приложения// Журнал Известия ЮФУ. Технические науки. – Ростов-на-Дону: Изд-во ЮФУ. – 2014 № 12 (161), С. 90-101.

3. Теплов, А.М. Об одном подходе к сравнению масштабируемости параллельных программ // Вычислительные методы и программирование: Новые вычислительные технологии (Электронный научный журнал). – М.: Изд-во МГУ, 2014.– Том 15, №4 – С. 697–711 (<http://num-meth.srcc.msu.ru/>).

##### **В других научных изданиях:**

4. Антонов, А.С., Воеводин, Вад. В., Воеводин, Вл. В., Теплов, А.М., Фролов, А.В. Первая версия Открытой энциклопедии свойств алгоритмов // Сборник «Параллельные вычислительные технологии (ПаВТ2015): труды международной научной конференции» (31 марта - 2 апреля 2015 г., г. Екатеринбург), Челябинск: Издательский центр ЮУрГУ, 2015.– С. 31-42.

5. Антонов, А.С., Теплов, А.М. Исследование масштабируемости программ с использованием инструментов анализа параллельных приложений на примере модели атмосферы NH3D // Вестник Южно-Уральского государственного ун-та / серия: Вычислительная математика и информатика. – Челябинск: Изд-во ЮУрГУ, 2013. – № 1, том 2.– С. 5 –16.

6. Антонов, А.С., Теплов, А.М. Исследование масштабируемости программ с использованием данных трассировщика и анализатора Intel Trace Analyzer and Collector на примере модели атмосферы NH3D // Сб. Научный сервис в сети Интернет: поиск новых решений. Труды Всероссийской науч-

ной конференции (17–22.09. 2012г., Новороссийск). – М.: Изд-во МГУ, 2012. – С. 121–129.

7. Антонов, А. С., Теплов, А.М. Использование данных системного мониторинга для определения факторов, уменьшающих масштабируемость приложения // Труды Всероссийской научной конференции «Научный сервис в сети Интернет: многообразие суперкомпьютерных миров» (г. Новороссийск, 22-27. 09. 2014 г.) – М.: Изд-во МГУ, 2014. – С. 87–96.

8. Антонов, А.С., Теплов, А.М. О практической сложности понятия масштабируемости параллельных программ // Сборник «Высокопроизводительные параллельные вычисления на кластерных системах (НРС 2014): Материалы XIV Международной конференции» (Пермь, 10-12 ноября 2014 г.). – Пермь: Изд-во Пермского национального исследовательского политехнического университета, 2014. – С. 20 – 27.

9. Фролов, А.В., Воеводин, В.В., Коньшин, И.Н., Теплов, А.М. Исследование структурных свойств алгоритма разложения Холецкого: от давно известных фактов до новых выводов // Сборник «Параллельные вычислительные технологии (ПаВТ'2015): труды международной научной конференции» (31 марта - 2 апреля 2015 г., г. Екатеринбург), Челябинск : Издательский центр ЮУрГУ, 2015 – С. 320-331.