

**Санкт-Петербургский политехнический университет
Петра Великого
Институт компьютерных наук и технологий**

На правах рукописи

Селин Иван Андреевич

**Методология балансировки нагрузки гибридных
высокопроизводительных систем**

Направление подготовки 09.06.01 «Информатика и вычислительная техника»

Направленность 09.06.01_01 «Математическое и программное обеспечение
вычислительных машин, комплексов и компьютерных сетей»

НАУЧНЫЙ ДОКЛАД

об основных результатах научно-квалификационной работы (диссертации)

Автор работы: Селин И.А.
Научный руководитель: доцент, к.т.н.,
Дробинцев П.Д.

Научно-квалификационная работа выполнена в ВШ/на кафедре Института компьютерных наук и технологий федерального государственного автономного образовательного учреждения высшего образования «Санкт-Петербургский политехнический университет Петра Великого»

Директор ВШ/зав. кафедрой: – *Дробинцев П.Д.,
доцент, к.т.н.*

Научный руководитель: – *Дробинцев П.Д.,
доцент, к.т.н.*

Рецензент: – *Голубев Алексей Андреевич
к.т.н., технический директор
ООО «ОРД»*

С научным докладом можно ознакомиться в библиотеке ФГАОУ ВО «Санкт-Петербургский политехнический университет Петра Великого» и на сайте Электронной библиотеки СПбПУ по адресу: <http://elib.spbstu.ru>

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность работы

Высокопроизводительные вычисления как таковые не возникли из ниоткуда. Сложность решаемых задач требовала всё больших ресурсов. Для примера можно рассмотреть типичную задачу из нефтегазовой отрасли – оптимизация процесса добычи нефти из подземного резервуара. Для оценки запасов нефти и определения необходимости бурения дополнительных скважин необходимо смоделировать процесс откачки по одним скважинам нефти и закачки воды по другим.

Для упрощения расчётов принимаем резервуар упрощённым до куба, этого будет достаточно для оценки количества производимых расчётов. Оптимальные размеры куба, которые гарантируют хоть сколько-нибудь правдоподобный результат, будут в районе 100 по каждой из координат, то есть $100*100*100$ точек. В каждой из этих точек необходимо вычислить вплоть до 30 функций: компоненты скорости, температуру, давление, концентрации различных веществ. Значения функций вычисляются как решения нелинейных уравнений, откуда следует что для каждого потребуется несколько сотен арифметических операций. А если исследуемый процесс является нестационарным, то потребуются дополнительные шаги по времени, обычно это сотни или тысячи. В итоге получается:

1000000 точек сетки * 15 функций * 500 операций * 500 временных шагов = $3.75 * 10^{12}$ арифметических операций.

Стоит отметить, что такое количество операций потребуется только лишь для одного относительно простого расчёта. Параметризация модели и отслеживание изменений входных данных потребует ещё большего количества ресурсов.

Очевидно, что нужды промышленности гораздо больше единичных расчётов. Нужно проводить огромное количество вычислительных экспериментов.

Но, какими бы мощными бы ни были суперкомпьютерные комплексы, их вычислительные способности ограничены. Поэтому разумно использовать имеющиеся ресурсы в максимально эффективном ключе.

Отсюда появляется потребность в совершенствовании существующих методов балансировки нагрузки. Речь идёт как про математические, алгоритмические методы, так и про модернизацию самих вычислительных комплексов, включая их гибридизацию, что позволяет лучше работать с разноплановыми задачами.

Учитывая натуру вычислительного эксперимента, он редко является атомарным. Чаще всего процесс расчёта возможно разделить на несколько частей, обладающих различными характеристиками.

Обладая современным гибридным вычислительным комплексом, состоящим из нескольких групп вычислительных узлов различного назначения, возможно распределить подзадачи на вычислительные узлы оптимальным образом.

Цель и задачи исследования

Цель работы – получение единой методики для оптимального размещения программного обеспечения на гибридных высокопроизводительных вычислительных комплексах

Для выполнения цели исследования были определены следующие задачи:

1. Определить основные ограничения запуска вычислительных задач на узлах определённого типа
2. Измерить целевой критерий целевого программного обеспечения в различных конфигурациях в рамках полученных ограничений

3. Разработать механизм по выбору оптимального размещения программного обеспечения на вычислительном комплексе на основе ранее проведённых экспериментов

Научная новизна

В работе представлены данные о вычислительных экспериментах работы различного программного обеспечения на различных конфигурациях гибридного суперкомпьютерного вычислительного комплекса на различных наборах входных данных для получения оценок производительности разноплановых задач.

На основе полученных данных были сформулированы основные ограничения вычислительных задач, потенциально влияющие на выбор подходящей архитектуры суперкомпьютерного комплекса для оптимального режима работы по различным критериям.

По данным экспериментов и сформулированным ограничениям была сформирована универсальная методика балансировки нагрузки гибридного высокопроизводительного вычислительного комплекса в зависимости от характера поступающей задачи.

Теоретическая и практическая значимость

Полученные результаты позволяют облегчить анализ целевого программного обеспечения, подлежащего запуску на гибридном высокопроизводительном вычислительном комплексе, предоставляя список ограничений, на которые стоит обратить внимание в первую очередь.

Разработанная методика позволяет в единообразном ключе рассматривать все задачи, поступающие для вычисления, и давать рекомендации по оптимальному конфигурированию гибридного вычислительного комплекса для каждой из них, что уменьшит трудоёмкость и внесёт элемент стандартизации в процессы адаптации программного обеспечения к современным суперкомпьютерным вычислительным комплексам.

Апробация работы

Результаты работы были представлены на:

4-й международной Уральской научной конференции по параллельным, распределённым и облачным вычислениям для молодых учёных (15 ноября 2018, Екатеринбург, «4th Ural Workshop on Parallel, Distributed, and Cloud Computing for Young Scientists», Yekaterinburg, Russia, November 15th, 2018),

4-й международной научно-практической конференции по инструментам и методам программной инженерии (3-4 марта 2017, Москва, «4th Tools & Methods of Program Analysis International Conference (TMPA-2017) organized in cooperation with ACM Sigsoft», Moscow, Russia, 3-4 March, 2017),

12-й Мультиконференции по проблемам управления (МКПУ-2019), 23-28 сентября 2019, Росси, Геленджик, Санаторий "Голубая даль",

10-й международной конференции для молодых учёных по тематике программной инженерии (30 мая – 1 июня, Россия, Москва, «Spring/Summer Young Researchers' Colloquium on Software Engineering 2016 (SYRCoSE-2016)», Moscow, Russia, May 30th – June 1st 2016),

9-й международной конференции по передовым методам производства и автоматизации (21-22 ноября 2019, Великобритания, Плимут, «9th International Workshop of Advanced Manufacturing and Automation» (IWAMA-2019), Plymouth, UK, November 21-22 2019).

Результаты работы были включены в отчёт по научно-исследовательской работе «Технологии и инструментарий для надежного управления производственными участками Интернета Вещей» (проект № 14.584.21.0022, идентификатор RFMEFI58417X0022).

Публикации

По результатам научно-квалификационной работы опубликовано 7 научных работ, в том числе 4 статьи, 3 статьи по материалам конференций в изданиях, 1 патент на изобретение, 2 свидетельства о регистрации программы для ЭВМ:

Представление научного доклада: основные положения

Были сформулированы основные ограничения для подзадач в рамках выполнения основной вычислительной задачи программного обеспечения, подлежащего запуску на гибридном высокопроизводительном вычислительном комплексе.

Предложена универсальная методика балансировки нагрузки для гибридного суперкомпьютера по поступающим на него задачам, опирающаяся на многочисленные данные вычислительных экспериментов.

СОДЕРЖАНИЕ РАБОТЫ

В работе исследовано влияние конфигурации гибридного высокопроизводительного вычислительного комплекса на итоговую производительность. Для этого на основе множественных вычислительных экспериментов были определены основные ограничения на по запуску различных типов задач на вычислительных узлах различных типов. Были замерены основные характеристики работы вычислительных задач для их оценки по целевому критерию. Разработан механизм по выбору оптимального размещения программного обеспечения на вычислительном комплексе на основе ранее проведённых экспериментов.

Объекты, (предмет) и методы исследования

Объектом исследования выпускной квалификационной работы является гибридная высокопроизводительная система и программное обеспечение, подлежащее запуску на ней с целью достижения максимальной производительности по указанному критерию.

Основной используемый метод в работе – проведение вычислительного эксперимента части рассматриваемого программного обеспечения с целью определения оптимальных параметров его конфигурации. В исследованиях применялись методы комбинаторики, теории автоматов, математической оптимизации, линейного программирования, дискретной математики, теории искусственного интеллекта и машинного обучения.

В процессе диссертационного исследования для экспериментов, расчётов и кодирования использовались такие инструментальные средства, как: Bourne Again SHell (bash), Simple Linux Utility for Resource Management (SLURM), Python, Java, IntelliJ IDEA.

Результаты и их обсуждение

В качестве тестового стенда использовался суперкомпьютер Санкт-Петербургского политехнического университета Петра Великого.

Суперкомпьютер «Политехник» является гибридной высокопроизводительной системой с пиковой производительностью более 1PFLOPS, разработанной компанией РСК [1]. Целью построения данного суперкомпьютера является повышение эффективности фундаментальных научных исследований в Санкт-Петербургском политехническом университете, обучение инженеров с высоким уровнем компетенции в суперкомпьютерных технологиях, установление регионального центра компетенций в области высокопроизводительных вычислений в наукоёмких областях экономики (биоинженерия, радиоэлектроника, авиастроение, энергомашиностроение).

Гибридный суперкомпьютер состоит из трёх частей: традиционного вычислительного кластера Tornado[2], системы с общей памятью Numascale[3] и кластера с x86-совместимыми ускорителями PetaStream[4] (Рис. 1).

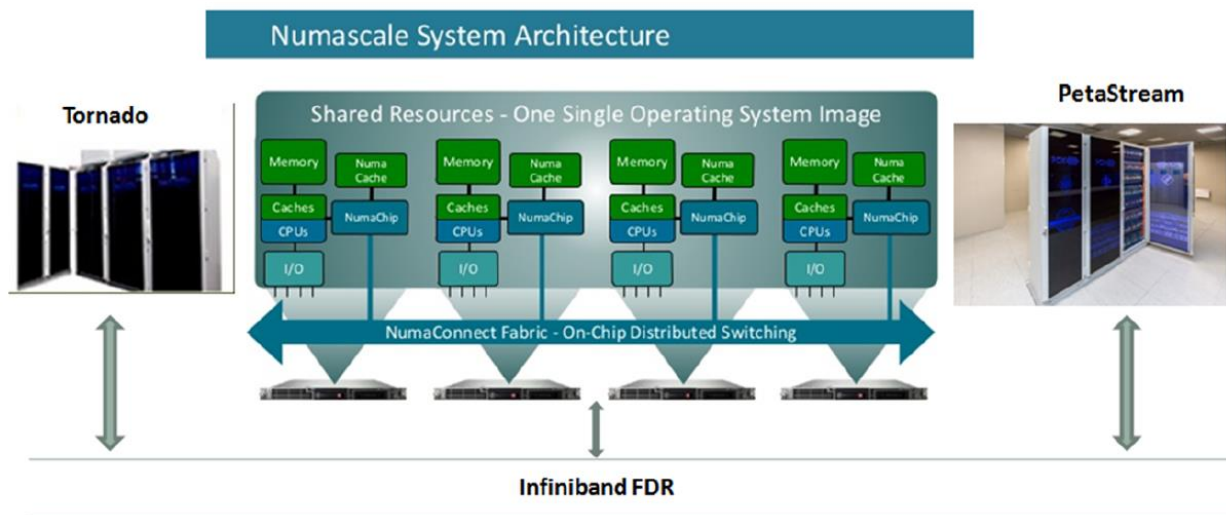


Рис. 1. Суперкомпьютер "Политехник"

Характеристики кластера Tornado: 668 вычислительных узлов с двумя центральными процессорами Intel E5-2697 v3, 64 ГБ оперативной памяти, сетевым соединением на основе технологии Infiniband (IB FDR). 56 из узлов

так же имеют по 2 ускорителя NVIDIA Tesla K40. Всего: 1336 ЦПУ, 18704 x86-ядер, 112 ускорителей, 42752 ГБ оперативной памяти.

Характеристики кластера Numascale: 64 вычислительных узла с тремя центральными процессорами AMD Opteron 6380 каждый, 192 ГБ оперативной памяти с кэш-когерентным неравномерным доступом (CC-NUMA), соединённые по топологии 3-мерного тора с помощью Infiniband (IB FDR), которые можно объединять в систему с общей памятью в различные конфигурации. Всего: 192 ЦПУ, 3072 x86-ядер, 12288 ГБ оперативной памяти.

Система хранения данных, состоящая из двух частей: параллельное хранилище Lustre [5] объёмом 1 ПБ и модульной системой хранения для облачных приложений, объёмом 0.5 ПБ.

Вычислительные эксперименты проводились на основе следующих проектов:

1. Фотограмметрическое программное обеспечение Agisoft Photoscan [6]
2. Инструмент тестирования на основе модели VRS/TAT [7,8]

Указанные проекты, как и любой другой сложны вычислительный процесс, характеризовались сложной структурой. Вычисления в рамках процесса делились на несколько этапов с различной степенью параллелизации, составом и количеством задач.

После интенсивного тестирования предложенного программного обеспечения были выявлены основные ограничения, являющиеся принципиальными препятствиями для размещения программного обеспечения на суперкомпьютерной инфраструктуре:

1. Ограничения по оперативной памяти. Были выявлены случаи, когда задачи не имели возможность завершиться в штатном режиме из-за нехватки оперативной памяти. Пример такой задачи можно увидеть на Рис. 2. Стоит отметить, что нагрузка имеет пиковый характер и большое количество памяти требуется на малую часть

вычислительного процесса, но ограничение вычисляется по максимальной нагрузке.

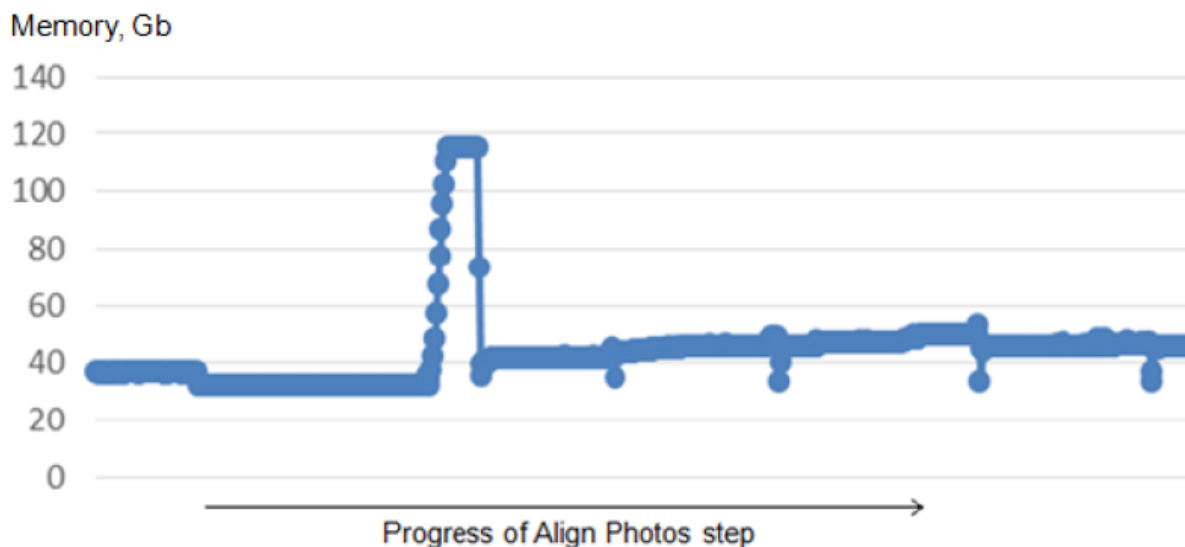


Рис. 2. Характер требований к памяти

2. Ограничения по долговременному хранилищу. Были выявлены случаи, когда задачи не имели возможность завершиться в штатном режиме из-за нехватки места на долговременном хранилище.
3. Ограничения по используемому набору инструкций центрального процессора и ускорителей. Были выявлены случаи, когда целевое программное обеспечение не могло работать на определённых вычислительных узлах из-за несоответствия набора инструкций в программе и в центральном процессоре или сопроцессоре.

Последнее ограничение ещё возможно обойти в случае наличия исходного кода и компиляции под указанную архитектуру. Но это возможно далеко не всегда, например, в случае проприетарных решений.

Так же были выделены факторы, влияющие на производительность задач:

1. Количество выделенных ресурсов.
2. Характеристики центрального процессора: их количество на каждом узле, количество ядер, тактовая частота, наличие или отсутствие

технологий одновременной многопоточности, расширенный набор инструкций.

3. Объём и частота оперативной памяти.
4. Наличие или отсутствие сопроцессора, ускорителя.
5. Скорость дисковых операций.
6. Скорость сетевых операций, сетевая топология.

Целевыми критериями с точки зрения выполнения задач, которые обычно используются в высокопроизводительных вычислениях, являются производительность в количестве операций в секунду, количество затраченной энергии, загруженность вычислительных узлов по различным показателям.

Однако, с пользовательской точки зрения гораздо более важными показателями являются итоговое время выполнения задачи и стоимость данных вычислений.

Время выполнения задачи в случае многостадийного расчёта определяется временем выполнения последней из подзадач. Соответственно, минимально возможное время задаётся характеристиками задачи в виде зависимости подзадач друг от друга и определяется критическим путём в сетевом графике зависимостей по данным, управлению.

Стоимость вычислений определяется количеством выделенных ресурсов во времени и измеряется в процессорном времени, умноженном на количество ядер, используемых в задаче.

Исходя из того, что критериев может быть несколько, измерять необходимо все показатели, задача подбора оптимальной конфигурации является многокритериальной.

Критерии оптимальности уже определены. Если задать их приоритет, можно попытаться определить оптимальное размещение программного обеспечения. Но как это сделать?

Благодаря таким технологиям, как OpenCL [9], HLS (High-Level Synthesis) [10] стало возможным кодировать программу один раз для множества различных архитектур.

Следующим шагом в данном отношении является создание механизма, который будет автоматически или автоматизированно определять оптимальный вычислитель. Для определения эффективности предлагается использовать уже озвученные критерии.

С точки зрения реализации, возможно 2 принципиально разных подхода:

1. С помощью исследования и валидации конкретного программного решения на конкретной высокопроизводительной системе
2. С помощью предиктивной аналитики

Первый метод (валидация) требует исполнения каждой отдельной части программы на всех типах вычислителей для точной оценки.

Предиктивные методы заключаются в анализе программного кода исследуемой программы с целью оценки, на каком из типов вычислителей она будет выполняться быстрее. Здесь не требуется исследование конкретной программы, но требуется обширная база знаний о том, какие программы в итоге на каких узлах запускались. То есть, исследование всё равно потребуется, но только на первичных этапах использования суперкомпьютерного комплекса, но не исследуемого программного обеспечения.

Оба метода не гарантируют точную оценку, т.к. в первом случае используются различные данные. А при различных данных поведение программы будет различным, вплоть до её бесконечного выполнения (см. проблема останова[11]). Во втором случае, обучение системы производится на основе данных по другим образцам программного обеспечения.

Валидационные испытания дают более точную оценку, но требуют больше трудозатрат для каждого образца адаптируемого программного обеспечения. Предиктивные же подходы, менее точны, но заметно быстрее с

точки зрения исполнения. Поэтому в зависимости от временных ограничений, стоит рассматривать оба варианта. Рекомендуется для долговременных проектов проводить валидационные испытания, а для краткосрочных – пользоваться предиктивными методами.

В случае первого метода уже есть оценки качества работы программного обеспечения в различных конфигурациях. Поэтому проблему оптимального выбора конфигурации можно свести пропорциям производительности различных подзадач на вычислительных узлах различного рода. Далее, указав принципиальные ограничения и сведя вместе пропорции, можно составить систему неравенств и свести задачу выбора оптимальной конфигурации гибридного вычислительного комплекса к многокритериальной задаче целочисленного линейного программирования с искомыми целочисленными переменными в качестве показателей использования ресурсов суперкомпьютерного комплекса различными видами подзадач на различных вычислительных узлах. Имея приоритетный список критериев, можно сформировать фронт подходящих решений и выбрать из ни наиболее подходящее.

Принципиально другой возможностью является использование предиктивных методов.

Анализ входящих событий на основе предиктивных методов можно реализовать несколькими способами:

1. Статический анализ программного обеспечения на основе анализа зависимостей по данным и по управлению
2. Статический анализ последовательности инструкций на основе нейросетевых технологий.
3. Динамический анализ сочетания последовательности инструкций и входных данных задачи, который стал возможен благодаря последним достижениям в сфере разработки нейросетевых технологий.

Статический анализ программного кода является достаточно изученной темой. Анализ же на основе нейросетевых анализаторов является темой в процессе становления. Последние исследования основываются на базе рекуррентных сетей долгой краткосрочной памяти (LSTM) [12], механизмов внимания (Attention) [13]. Модули динамического анализа ещё находятся в активной разработке и ещё будут улучшаться.

В зависимости от входных данные, поведение программы может значительно отличаться, поэтому динамический анализ может привести значительные улучшения для задач, которые могут быть плохо обусловлены (но не всегда плохо обусловлены).

Резюмируя вышесказанное, методология балансировки на высоком уровне может быть описана следующей диаграммой (Рис. 3):

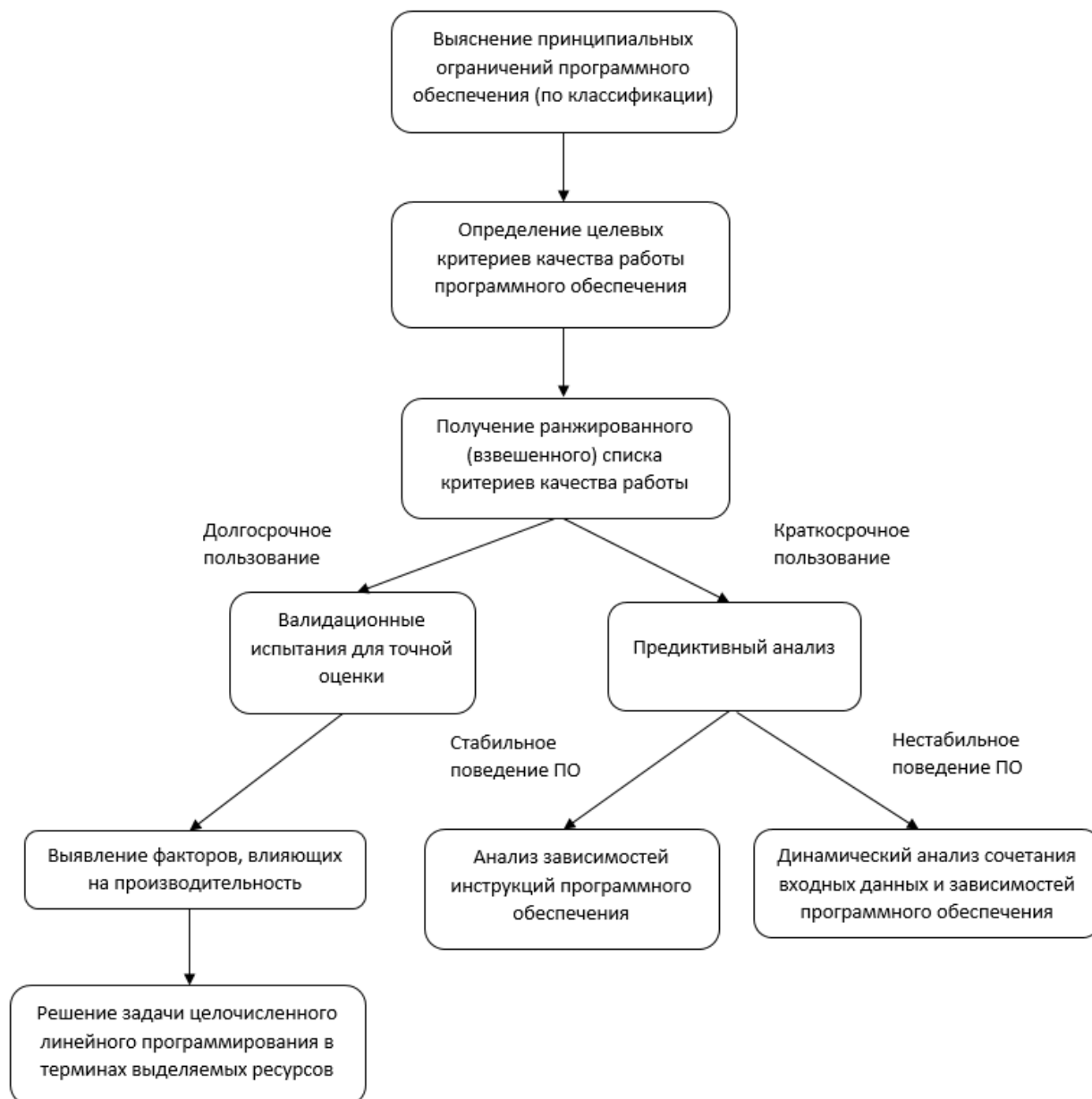


Рис. 3. Методология балансировки нагрузки

По разработанной методологии был осуществлён запуск проекта фотограмметрического программного обеспечения Agisoft Photoscan в распределённом режиме [14].

Было обнаружено принципиальное ограничение:

1. Экспоненциальный рост количества необходимой оперативной памяти от количества обрабатываемых фотографий на этапе выравнивания фотографий (Match Photos) (Рис. 4) и при размере проекта более 50 тысяч фотографий классической кластерной архитектуры уже не хватало. Для проекта в 75 тысяч фотографий пришлось дополнить список вычислительных узлов одним узлом из

NUMA-кластера, т.к. он обладал достаточным количеством оперативной памяти для выполнения задачи. Однако для проектов в 100 тысяч фотографий и более не хватило и такой конфигурации. Поэтому для таких входных данных требуется организация NUMA-кластера.

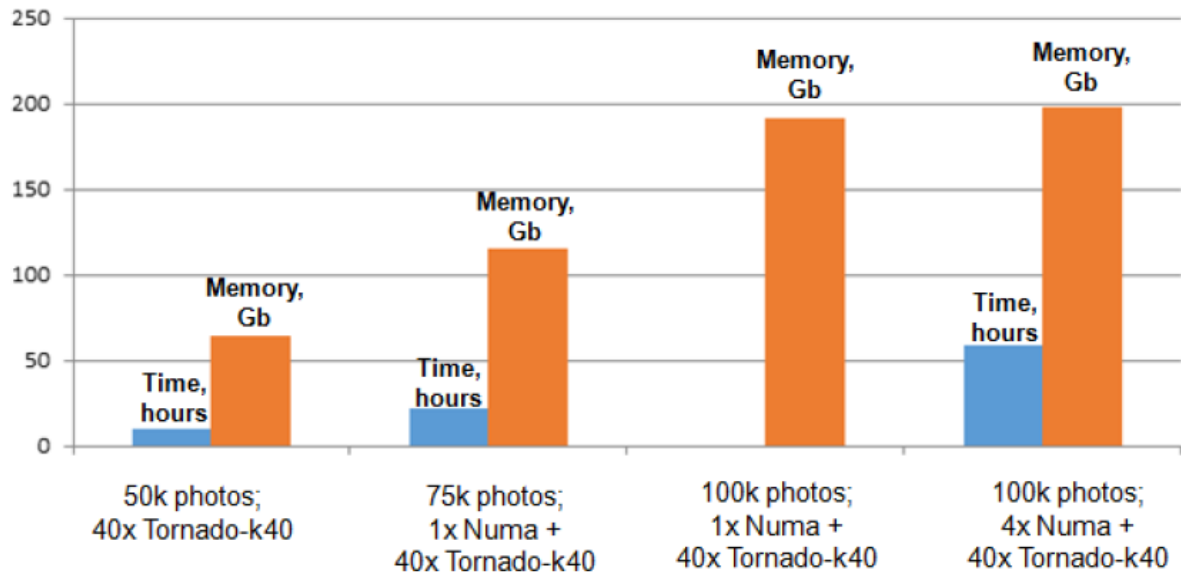


Рис. 4. Задача выравнивания камер с существенным ограничением по памяти

Табл. 1. Анализ производительности Agisoft Photoscan на вычислительных узлах различного типа на проектах малого размера

Step	8xNUMA	8xTornado	8xTornado-k40
Match Photos	2:29	1:13	1:13
Align Photos	1:22	0:32	0:32
Build Dense Cloud	11:07	6:39	1:58
Build DEM	0:19	0:10	0:10
Build Orthomosaic	0:50	0:33	0:33
Overall	16:07	9:07	4:26

Были проанализированы факторы, влияющие на производительность (Табл. 1):

1. В целом, традиционная кластерная архитектура (на основе узлов Tornado) показала более высокую производительность, чем архитектура с общей памятью (Numascale). Узлы с общей памятью показали результаты в 1,7-4 раза хуже, чем узлы Tornado.

2. Наличие GPGPU-ускорителей играет роль только при построении плотного облака точек (Build Dense Cloud), но приносит значительный прирост производительности
3. Дополнение традиционных вычислительных узлов NUMA-узлами немного снижает общую производительность за счёт большего времени выполнения задачи на узле Numascale, но позволяет решать гораздо более широкий спектр задач.

Вместе с получением ограничений и факторов, влияющих на производительность, были выделены критерии качества работы программного обеспечения, которые заключались в первую очередь в быстродействии. Поэтому целевой метрикой было общее время выполнения всего многостадийного вычислительного процесса.

Основываясь на этой метрике, были разработаны следующие правила для организации последующих вычислительных экспериментов (в зависимости от объема входных данных проектов):

1. До 50 тысяч фотографий – узлы Tornado везде, кроме построения плотного облака точек. Там стоит использовать узлы Tornado-k40 с GPGPU-ускорителями
2. До 75 тысяч фотографий – узлы Tornado и один узел Numascale везде, кроме построения плотного облака точек. Там стоит использовать узлы Tornado-k40 с GPGPU-ускорителями
3. 100 тысяч фотографий и больше – узлы Tornado с объединёнными узлами Numascale в узел системы с общей памятью, кроме построения плотного облака точек. Там стоит использовать узлы Tornado-k40 с GPGPU-ускорителями
4. Для экстра-больших проектов стоит применять только узлы Numascale.

Список литературы:

1. Суперкомпьютерный центр «Политехнический». <https://scc.spbstu.ru>

2. Creating "Polytechnic RSC Tornado" supercomputer for St. Petersburg State Polytechnical University. <http://www.rscgroup.ru/ru/our-projects/240-sozdanie-superkompyutera-politehnik-rsk-tornado-dlya-spbpu>
3. Einar Rustad. NumaConnect White Paper: A high level technical overview of the NumaConnect technology and products. https://www.numascale.com/numa_pdfs/numaconnect-white-paper.pdf
4. Creating "Polytechnic RSC PetaStream" supercomputer for St. Petersburg State Polytechnical University. <http://www.rscgroup.ru/ru/our-projects/242-sozdanie-superkompyutera-politehnik-rsc-petastream-dlya-spbpu>
5. Lustre File System. <https://lustre.org/about/>
6. Agisoft PhotoScan. <http://www.agisoft.com>
7. Веселов А.О., Иванов А.С., Котляров В.П., Тютин Б.В., Автоматизации тестирования телекоммуникационных приложений, 2009.
8. Воинов Н.В. Методы генерации тестовых сценариев на основе структурированных UCM-моделей проектируемой системы
Диссерт. на соискание уч. ст. канд. тех. наук. СПб.: СПбГПУ, 2011.
170 с
9. John E. Stone, David Gohara, and Guochun Shi. OpenCL: A Parallel Programming Standard for Heterogeneous Computing Systems. Comput Sci Eng. 2010 May; 12(3): 66–72
10. Philippe Coussy, Adam Morawiec. High-Level Synthesis: from Algorithm to Digital Circuit. Springer Publishing Company, Incorporated, 300pp. ISBN:978-1-4020-8587-1
11. A. Turing. On Computable Numbers With an Application to the Entscheidungsproblem. 1937 (reprinted in Davis, 1965, p. 115)
12. Sepp Hochreiter; Jürgen Schmidhuber (1997). "Long short-term memory". Neural Computation. 9 (8): 1735–1780.

13. K. Cho, et al. Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), October, 2014
14. An approach for managing hybrid supercomputer resources in photogrammetric tasks Voinov, N., Selin, I., Drobintsev, P., Kotlyarov, V. 2018 CEUR Workshop Proceedings 2281, pp. 12-19

Заключение

Таким образом, в рамках диссертационной работы были сформулированы возможные ограничивающие факторы для запуска вычислительных задач определённого типа. Была приведены возможные критерии оценки качества работы программного обеспечения и методы работы с ними. На основе полученных данных была разработана методология балансировки нагрузки гибридных высокопроизводительных систем.

Полученные результаты имеют значительное практическое значение и позволяют упростить и стандартизировать процесс вывода программного обеспечения на гибридные высокопроизводительные системы.

**Список работ, опубликованных по теме научно-квалификационной
работы (диссертации)**

Публикации в изданиях, рецензируемых ВАК

1. ПРЕОБРАЗОВАНИЕ АБСТРАКТНЫХ ПОВЕДЕНЧЕСКИХ СЦЕНАРИЕВ В СЦЕНАРИИ ПРИМЕНИМЫЕ ДЛЯ ТЕСТИРОВАНИЯ
Дробинцев П.Д., Котляров В.П., Никифоров И.В., Воинов Н.В., Селин И.А.
Труды Института системного программирования РАН. 2016. Т. 28. № 3. С. 145-160.
2. ГЕНЕРАЦИЯ ТЕСТОВОГО НАБОРА НА ОСНОВЕ ПОТОКА УПРАВЛЕНИЯ
Воинов Н.В., Дробинцев П.Д., Котляров В.П., Никифоров И.В., Селин И.А.
Системы и средства информатики. 2015. Т. 25. № 1. С. 54-73.

Публикации в других изданиях

1. Technology and tools for developing industrial software test suites based on formal models and implementing scalable testing process on supercomputer
Kotlyarov, V., Drobintsev, P., Voinov, N., Selin, I., Tolstoles, A. 2018
Communications in Computer and Information Science 779, pp. 51-63
2. An approach for managing hybrid supercomputer resources in photogrammetric tasks
Voinov, N., Selin, I., Drobintsev, P., Kotlyarov, V.
2018 CEUR Workshop Proceedings 2281, pp. 12-19
3. Building a time-optimal power consumption strategy for a solar car. Selin, I.A., Kasatkin, I.I., Zakhlebaev, E.A., Hemminger, O.P. 2019. IOP Conference Series: Materials Science and Engineering 643(1),012004
4. Optimization of Technological Processes at Production Sites Based on Digital Modeling
Drobintsev, P., Voinov, N., Kotlyarova, L., Selin, I., Aleksandrova, O. 2020 Lecture Notes in Electrical Engineering 634 LNEE, pp. 600-607
5. Захлебаев Е.А., Селин И.А., Касаткин И.И. ПРОГРАММА ДЛЯ ВЫЧИСЛЕНИЯ ОПТИМАЛЬНОГО ПЛАНА ИСПОЛЬЗОВАНИЯ

ЭЛЕКТРОМОБИЛЯ НА СОЛНЕЧНЫХ ПАНЕЛЯХ НА ОСНОВЕ
ПРОГНОЗА О ПРИХОДЯЩЕЙ И РАСХОДУЕМОЙ ЭНЕРГИИ.

Свидетельство о регистрации программы для ЭВМ RU 2019665646,
27.11.2019. Заявка № 2019664436 от 15.11.2019.

6. Селин И.А. ОЦЕНИВАНИЕ И ОПТИМИЗАЦИЯ СТРАТЕГИИ
ЭНЕРГОСБЕРЕЖЕНИЯ ДЛЯ СОЛНЦЕМОБИЛЯ. В сборнике: Материалы
XII мультikonференции по проблемам управления (МКПУ-2019). В четырех
томах. 2019. С. 213-216.

Аспирант _____ Селин И.А.

(подпись)