

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
ПРИ СОДРУЖЕСТВЕ НГТУ И НГУ



**НГТУ
НЭТИ**

N* Новосибирский
государственный
университет
***НАСТОЯЩАЯ НАУКА**

**Разработка и реализация алгоритмов динамической балансировки
вычислительной нагрузки для подсистемы воспроизведения трасс в
системе LuNA**

Автор выпускной квалификационной работы: Саяпин М. П.
Научный руководитель: Малышкин В. Э., д.т.н., профессор.

Новосибирск 2022

Введение

- Разработка систем для автоматического построения высокопроизводительных распределенных программ включает в себя такие задачи, как:
 - динамическая балансировка нагрузки,
 - обеспечение коммуникаций между вычислительными узлами,
 - настройка под имеющиеся аппаратные ресурсы.
- Разработка таких средств является сложной задачей в общем случае, что зачастую выливается в меньшую эффективность конструирования программы.
- Одним из таких средств является система фрагментированного программирования LuNA.
- Возможное решение проблемы снижения эффективности программы – это оптимизация на основе воспроизведения трасс.
- Такой подход не всегда выгоден, потому что при воспроизведении трассы или изменении входных данных, может возникать дисбаланс вычислительной загрузки.

Анализ предметной области

Критерии для обзора систем параллельного программирования:

- Оценка времени выполнения задач для определения уровня нагрузки задачи.
- Оценка объёма предстоящей работы по выполнению задачи
- Выявление информационных зависимостей между задачами

Системы параллельного программирования:

- Charm++
- Legion
- DVM-H
- LuNA

Вывод: воспроизведение с динамической балансировкой вычислительной нагрузки трассы программы является недостаточно проработанной темой, исследование которой является целесообразным.

Цель и задачи

Цель работы – разработать и интегрировать в проигрыватель трасс динамический балансировщик вычислительной нагрузки.

Задачи:

- Разработка или адаптация алгоритма динамической балансировки вычислительной нагрузки
- Разработка и реализация алгоритма перераспределения вычислительной нагрузки по узлам
- Реализация и доработка алгоритма динамической балансировки вычислительной нагрузки
- Экспериментальные исследования работы динамического балансировщика вычислительной нагрузки

Необходимые определения

Система фрагментированного программирования LuNA –
задачеориентированная система параллельного программирования.

- Фрагмент данных (ФД) – набор переменных.
- Фрагмент вычислений (ФВ) – программная единица $\langle in, mod, out \rangle$, содержащая описание входных, выходных фрагментов данных и исполняемого кода фрагмента.
- Вычислительная нагрузка – выполняемая программная единица.
- Вычислительная загрузка – процессорное время вычислительной единицы.
- Трассировка – запись событий работы программы.
- Трасса – результат трассировки, хранящийся в файле.
- Воспроизведение трассы – это считывание и воспроизведение трассы, содержащую в себе инструкции по запуску ФВ и передаче ФД.

Действующий алгоритм

Трасса воспроизводится под управлением среды исполнения пулом рабочих потоков на узлах выполнения.

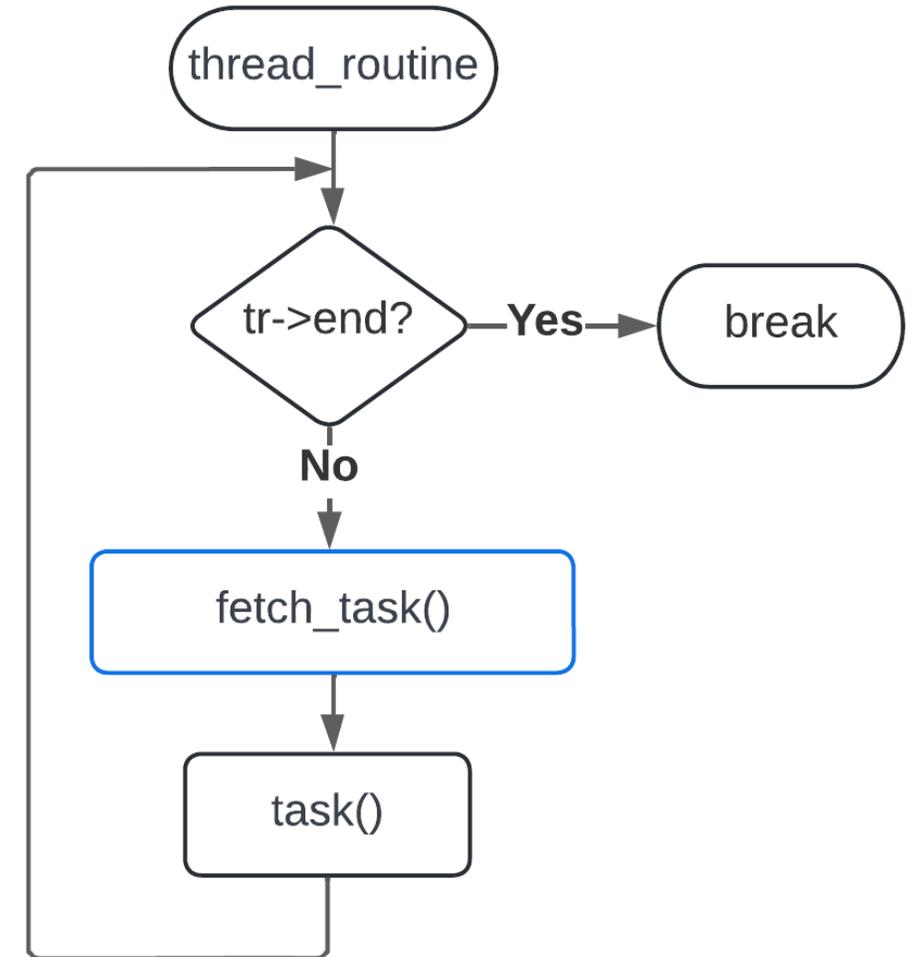
Плюсы

- Малые накладные расходы
- Отсутствие сборщика мусора

Минусы:

- Смена конфигурации оборудования
- Изменение входных данных
- Принятые решения системой LuNA

Предлагается добавить динамический балансировщик вычислительной нагрузки в проигрыватель трасс.



Выбор алгоритма динамического балансировщика

Требования, предъявляемые к алгоритму:

- Малое количество накладных расходов по сравнению с проигрывателем трасс без динамического балансировщика вычислительной нагрузки
- Улучшение эффективности выполнения программы на некоторых задачах по сравнению с действующим алгоритмом, при наличии простаивающих вычислительных мощностей
- Параметризация алгоритма

Было решено взять за основу и адаптировать конкретным условиям алгоритм динамической балансировки вычислительной нагрузки WorkStealing¹.

1 Cederman D., Tsigas Ph. Dynamic Load Balancing Using Work-Stealing // GPU Computing Gems Jade Edition. 2011. P. 485–500.

WorkStealing

- Суть алгоритма WorkStealing заключается в том, что с определённой периодичностью или по событию узлы проверяют свою загруженность и, если в результате проверки какой-либо узел оказывается недогружен, он отправляет запросы на получение работы на другие узлы.
- Когда узел получает запрос на выделение работы, он должен принять решение о её выделении, основываясь, в свою очередь, на собственной перегруженности. Тем самым достигается выравнивание вычислительной нагрузки

Вопросы для рассмотрения:

- Вычисление степени загрузки узла
- Обнаружение дисбаланса загрузки
- Передача по сети фрагментов вычислений и данных

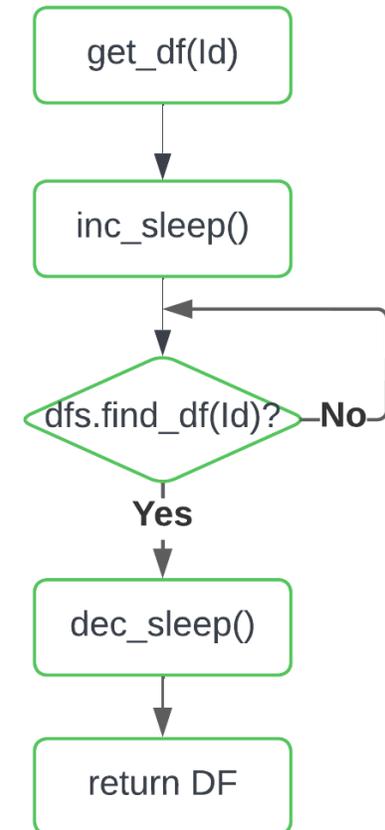
Степень и дисбаланс загрузки узла

Определение степени загрузки узла:

- Принято решение измерять уровень вычислительной загрузки узла, как количество спящих потоков из пула рабочих потоков узла

Обнаружение дисбаланса:

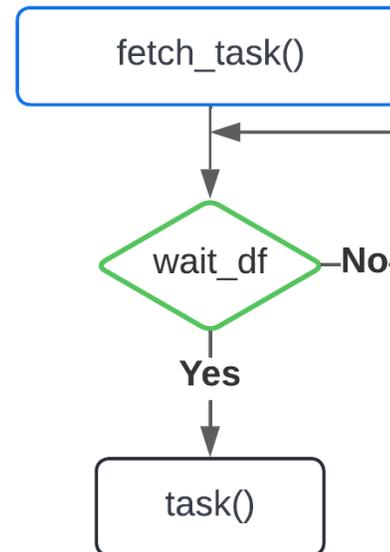
- При запросе или получении ФД потоком узла, проверяется текущая степень загрузки узла
- При недогрузке или перегрузке узла отправляется соответствующее сообщение другим узлам
- При условии окончания трассы узлы также оповещаются
- При высокой загруженности узла и наличии недогруженных узлов ФВ отправляется недогруженному узлу



Передача по сети ФД и ФВ

Разработан и реализован алгоритм по переброске ФВ и ФД по узлам.

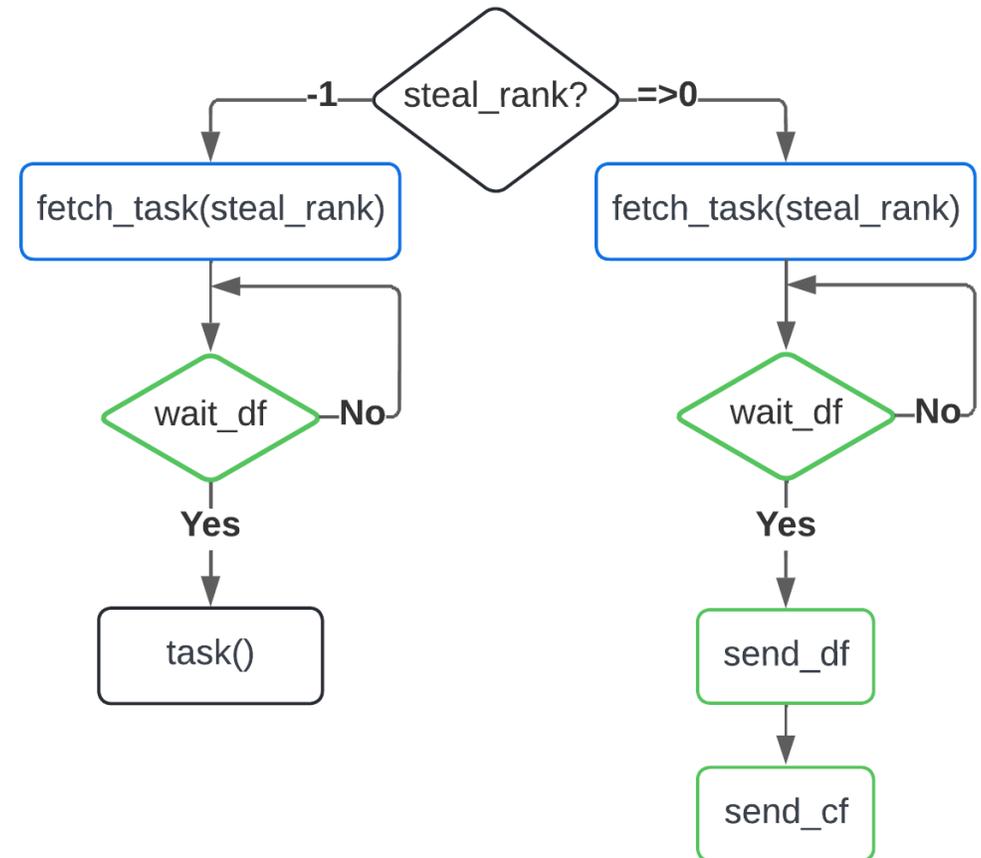
- При возникновении дисбаланса вычислительной загрузки ФВ на перегруженном узле считывается из трассы для передачи по сети
- Узел ожидает необходимые ФД для ФВ, и отправляет ФД на узел-получатель
- После этого отправляется сообщение по сети с ФВ узлу-получателю



Передача по сети ФД и ФВ

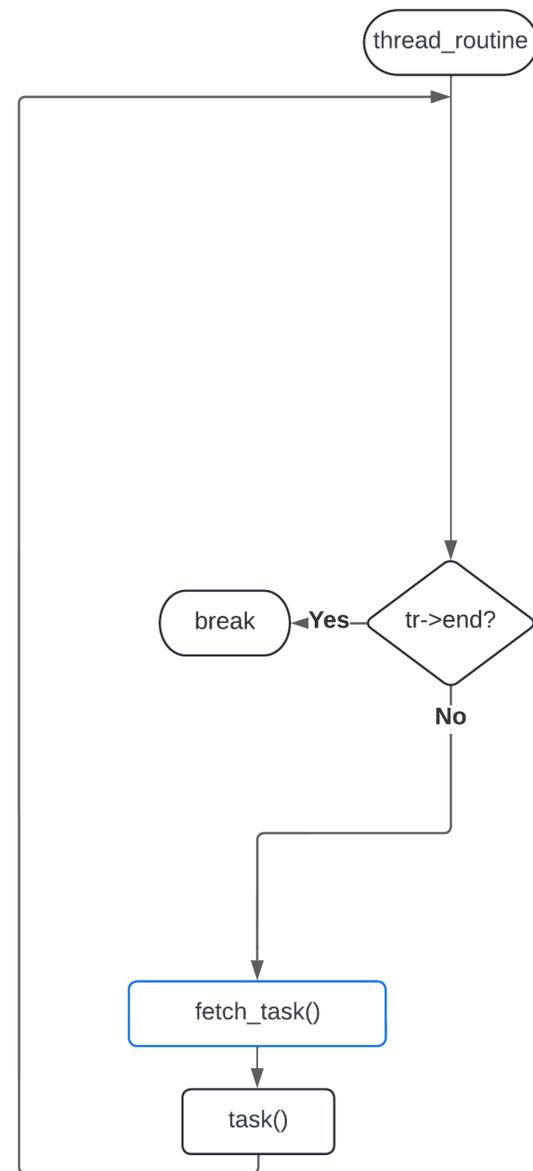
Разработан и реализован алгоритм по переброске ФВ и ФД по узлам.

- При возникновении дисбаланса вычислительной загрузки ФВ из трассы считывается для передачи по сети
- Узел ожидает необходимые ФД для ФВ, и отправляет ФД на узел-получатель
- После этого отправляется сообщение по сети с ФВ узлу-получателю



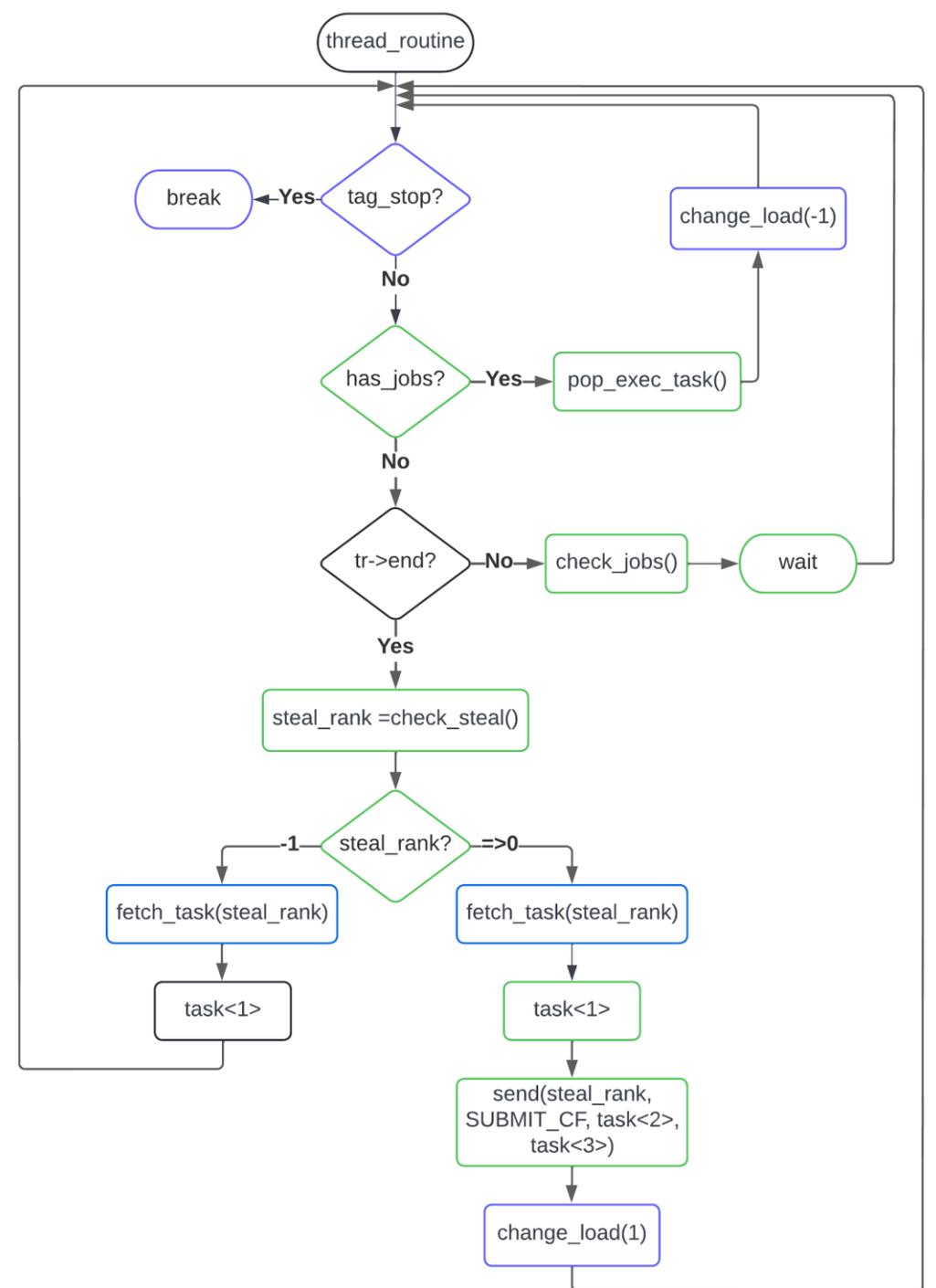
Модифицированный алгоритм

- Учтены особенности считывания ФВ из трассы для передачи по сети
- Реализована передача ФД по сети
- Добавлены алгоритмы для вычисления степени загрузки узла
- Добавлены алгоритмы для выявления дисбаланса вычислительной загрузки
- Изменена схема остановки работы программы, поскольку после окончания трассы на узле возможна передача по сети ФВ
- Система завершает работу только после передачи и выполнения всех ФВ

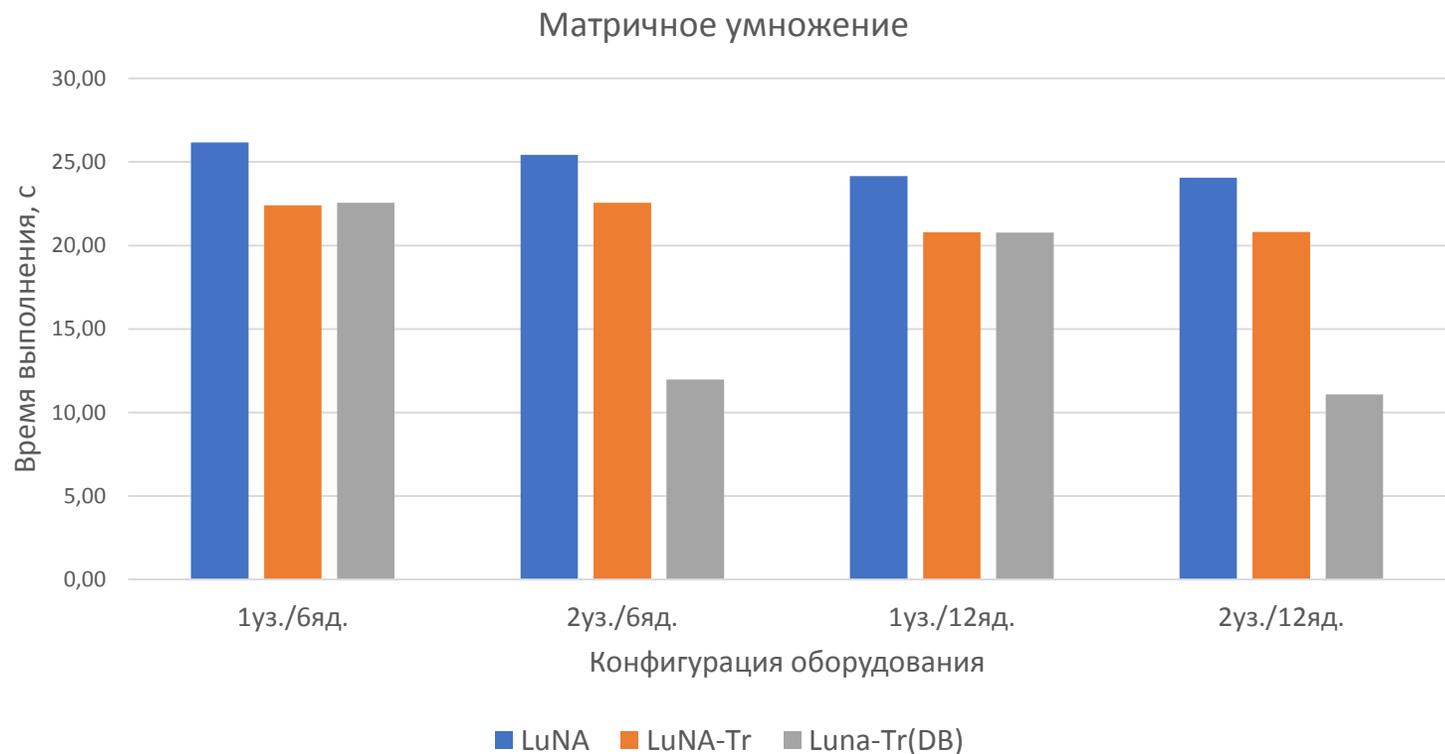


Модифицированный алгоритм

- Учтены особенности считывания ФВ из трассы для передачи по сети
- Реализована передача ФД по сети
- Добавлены алгоритмы для вычисления степени загрузки узла
- Добавлены алгоритмы для выявления дисбаланса вычислительной загрузки
- Изменена схема остановки работы программы, поскольку после окончания трассы на узле возможна передача по сети ФВ
- Система завершает работу только после передачи и выполнения всех ФВ



Исследования



- Результатом тестирования на 1 узле является идентичное время выполнение программы с помощью подсистемы LuNA-Trace(DB) по сравнению с LuNA-Trace
- Результатом тестирования на 2 узлах является улучшение времени выполнения программы LuNA-Trace-DB по сравнению с LuNA-Trace в ~1.9 раз

Вывод: результаты проведенных тестирований указывают на небольшие накладные расходы динамического балансировщика вычислительной нагрузки в подсистеме LuNA-Trace(DB), а также на распределение динамическим балансировщиком вычислительной нагрузки по узлам.

Заключение

В результате работы была обеспечена динамическая балансировка вычислительной нагрузки в подсистеме проигрывателя трасс, в частности разработан и реализован алгоритм динамической балансировки вычислительной нагрузки, который:

- определяет уровень вычислительной загрузки узла
- выявляет дисбаланс вычислительной загрузки на узлах
- распределяет вычислительную нагрузку по узлам.

Реализованный алгоритм был интегрирован в подсистему проигрывателя трасс LuNA.

Было проведено тестирование, показавшее работоспособность алгоритма по устранению дисбаланса вычислительной загрузки.

Направления дальнейшей работы

- В дальнейшем планируется работа по усовершенствованию алгоритма динамической балансировки вычислительной нагрузки путем более полного использования информации, содержащейся в трассе.

Благодарность

Хочу выразить глубокую признательность научному сотруднику Института вычислительной математики и математической геофизики Перепёлкину Владиславу Александровичу, за чуткое руководство и помощь при подготовке выпускной квалификационной работы.

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
ПРИ СОДРУЖЕСТВЕ НГТУ И НГУ



**НГТУ
НЭТИ**

N* Новосибирский
государственный
университет
***НАСТОЯЩАЯ НАУКА**

**Разработка и реализация алгоритмов динамической балансировки
вычислительной нагрузки для подсистемы воспроизведения трасс в
системе LuNA**

Автор выпускной квалификационной работы: Саяпин М. П.
Научный руководитель: Малышкин В. Э., д.т.н., профессор.

Новосибирск 2022