

# **Разработка и реализация алгоритмов исполнения фрагментированных программ с заданным поведением**

Выполнил: Тренин С. А.  
ФИТ НГУ, гр. 15205

Руководитель: д.т.н., проф. Малышкин В. Э.  
ИВМиМГ СО РАН

Новосибирск 2019



# Введение

Высокоуровневое программирование актуально в численном моделировании за счёт снижения порога вхождения в технологии, но для широкого класса задач малоприменимо в силу невысокой производительности.

Основная причина невысокой производительности - фундаментальные проблемы выбора и реализации выбранного поведения.



# Обзор существующих решений

Применяются следующие подходы:

- 1) Введение в язык специальных операторов, эвристик и директив, задающих прямое управление. При этом, уровень программирования снижается (SMP Superscalar, ProActive Parallel Suite).
- 2) Сужение предметной области, на которую ориентирована система (PaRSEC, TBLAS).
- 3) Иногда в системах применяют оба вышеуказанных подхода (LuNA, Legion, Julia).



## Обзор существующих решений

Дальнейшее развитие специализированных операторов и директив вносит вклад в автоматизацию выбора поведения.

Расширение же области применения специализированных языков, как видно на примере Julia, порождает в нём новые специализированные операторы.

Проблема выбора наилучшего поведения в полностью не решена ни в одной из существующих систем. Таким образом, разработка новых средства управления поведением и поддерживающих их алгоритмов остаётся актуальной задачей.



# Система LuNA

Позволяет автоматизировать генерацию параллельных программ при помощи технологии фрагментированного программирования. Управление поведением вынесено отдельно (изменение поведения не затрагивает функциональные свойства программы) и автоматизировано, что делает систему удобной для исследования управления поведением.



## Постановка цели и задач

Цель - разработка средств задания поведения и системных алгоритмов для системы LuNA, поддерживающих эти средства.

### Задачи

- 1) Выбрать средств управления поведением во фрагментированных программах.
- 2) Разработать алгоритмы, реализующие выбранные средства в системе LuNA.
- 3) Провести эксперименты на конкретных тестах.



## Необходимые определения

В технологии фрагментированного программирования (ФП) исполняющаяся программа состоит из множества последовательных подзадач, связанных информационными связями.

Фрагмент данных (ФД) - переменная единственного присваивания.

Фрагмент вычислений (ФВ) - операция единственного срабатывания.



## Предлагаемое решение

В результате анализа были выявлены основные направления для фиксации поведения:

- 1) Распределением фрагментов вычислений по вычислительным узлам.
- 2) Динамическая балансировка нагрузки.
- 3) Сборка мусора.
- 4) Выделение и распределение ресурсов.





## Динамическая балансировка нагрузки

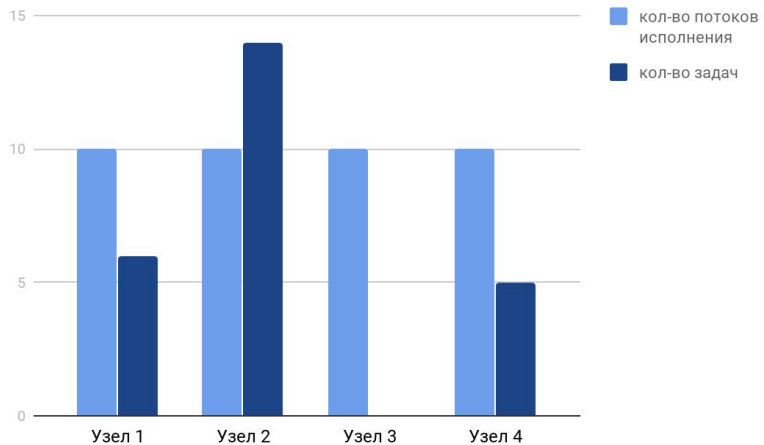
На данный момент в системе LuNA реализована балансировка методом Rope of Beads, который является алгоритмом диффузионного типа с линейной структурой. Такая балансировка хорошо устраняет общий дисбаланс, но плохо подходит для устранения точечного дисбаланса.

В результате анализа существующих методов балансировки был выбран метод Work Stealing, так как он относится к другому типу балансировок и устраняет точечный дисбаланс лучше чем Rope of Beads.

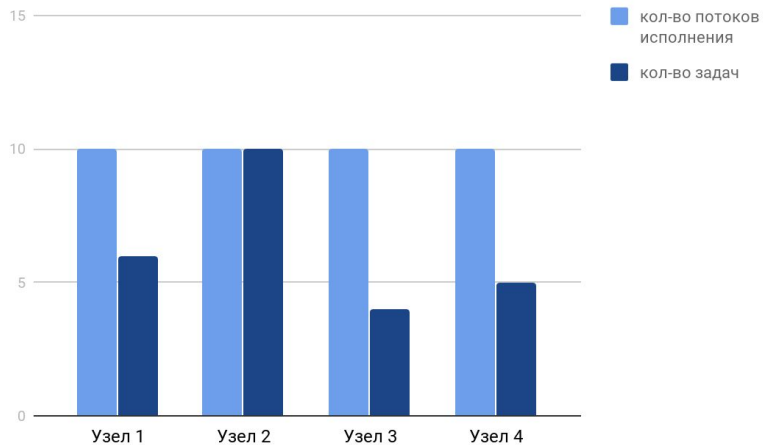


# Метод Work Stealing

Без балансировки



С балансировкой





## Управление распределением ресурсов

На данный момент управляющий узел рассылает готовые к исполнению ФВ вместе в их ФД на узлы-вычислители, которые, в свою очередь возвращают вычисленные ФД без сохранения.

Решено добавить:

1. Кэширование ФД на вычислительных узлах.
2. Перевычисление ФД из ФВ на узле вместо посылки ФД, когда это ВЫГОДНО.



# Порядок исполнения фрагментов вычислений

На данный момент в системе LuNA порядок вычислений устанавливается информационными зависимостями.

Решено добавить:

1. Систему приоритетов для ФВ (по умолчанию 0, можно задавать рекомендациями). При прочих равных в рамках узла первым исполниться ФВ с большим приоритетом.
2. Установку строго порядка исполнения ФВ путём добавки фиктивного ФД, передаваемого из одного ФВ в другой.



## Список предложенных средств

- 1) Распределением фрагментов вычислений по вычислительным узлам.
  - a) Приоритеты
  - b) строгий порядок
- 2) Динамическая балансировка нагрузки.
  - a) Work Stealing
- 3) Сборка мусора.
- 4) Выделение и распределение ресурсов.
  - a) Кэширование
  - b) Перевычисление

Таким образом, мной были выбраны три основных пути фиксации поведения, ещё не реализованный в системе LuNA, наиболее важные для дальнейшего исследования вопросов задания и исполнения поведения фрагментированных программ.



# Тестирование

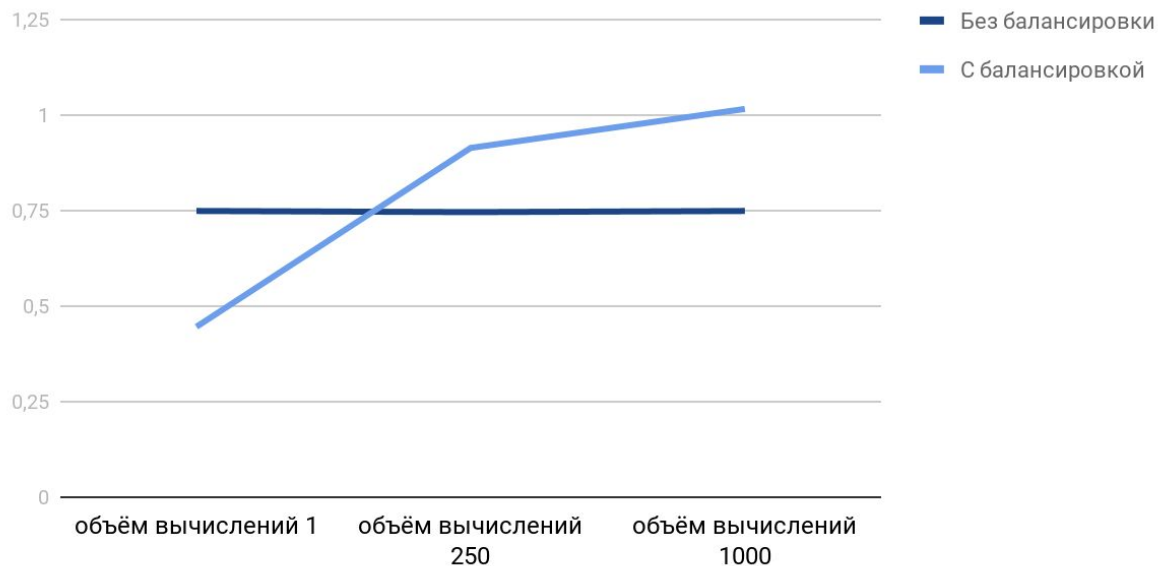
Для тестирования введённой динамической балансировки нагрузки по методу WorkStealing была взята программа на языке LuNA, решающая модельную задачу, реализующую итерационный метод с параметризованным объёмом вычислений в узлах сетки.

Запуск производился на кластере МВС-10П Межведомственного Суперкомпьютерного Центра. Задача запускалась на 4 вычислительных узлах с 2 потоками исполнения на каждом. При этом, чтобы наглядно продемонстрировать эффект динамической балансировки, один из узлов изначально не был загружен работой.

# Тестирование

Как можно видеть, в данный момент реализованный алгоритм WorkStealing в рамках данной задачи лучше всего показывает себя при большом объёме вычислений в узлах сетки.

Отношение теоретического времени работы (при идеальной балансировке) к реальному





## Публикации по теме

Публикации по теме:

- 1) Участие в летней международной XXXI молодежной Школе-конференции по параллельному программированию г. Новосибирск
- 2) Выступление на Международной научной студенческой конференции 2019





# Заключение

## Защищаемые положения

- 1) На основании анализа были выбраны средства управления поведением во фрагментированных программах.
- 2) Разработаны алгоритмы, реализующие выбранные средства в системе LuNA.
- 3) Проведены эксперименты на конкретных тестах.

В результате проведённой работы на базе системы LuNA были разработаны средства задания поведения и добавлены системные алгоритмы, поддерживающие эти средства.

Далее планируется улучшение выбранных алгоритмов на основе профилирования. Кроме того, планируется дальнейший подбор параметров балансировки (возможно с дальнейшей автоматизацией на основе профилирования).