ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«НОВОСИБИРСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
(НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ, НГУ)
Факультет информационных технологий
Кафедра параллельных вычислений
09.03.01 Информатика и вычислительная техника. Программная инженерия и компьютерные науки

Разработка эффективной фрагментированной программы решения задачи фильтрации двухфазной жидкости

Выполнил: Спирин Виталий Андреевич, студент группы 19203 ФИТ НГУ

Научный руководитель: Малышкин Виктор Эммануилович, д.т.н., профессор, зав. каф. ПВ ФИТ НГУ

Соруководитель: Перепёлкин Владислав Александрович, ст. преп. каф. ПВ ФИТ НГУ

Рецензент: Дучков Антон Альбертович, к.ф.-м.н., зав. лаб. ИНГГ СО РАН

Новосибирск 2023

22.06.2023

Актуальность

В области численного моделирования на мультикомпьютерах стоит проблема разработки *эффективных* параллельных программ.

- по времени работы
- по расходу памяти
- по нагрузке на сеть
- …

Здесь требуется специальная квалификация в области системного параллельного программирования:

- декомпозиция данных и вычислений
- балансировка нагрузки
- отладка коммуникаций
- …

Можно использовать ***системы с автоматизацией конструирования параллельных программ***, которые выполняют конструирование по высокоуровневым спецификациям.

Актуальность

Задача автоматического конструирования эффективных параллельных программ по их высокоуровневым спецификациям *алгоритмически труднорешаема*, поэтому в системах используются частные алгоритмы и эвристики.

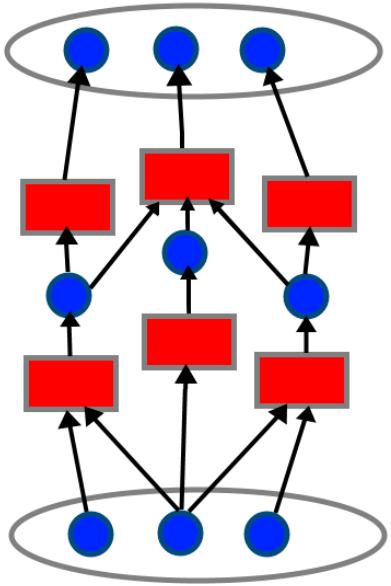
Для улучшения таких систем требуется исследовать ***ручную*** разработку эффективных параллельных программ ***для частных случаев*** среди задач численного моделирования, ***чтобы накапливать и систематизировать опыт***.

Данная работа посвящена такому исследованию для задачи моделирования процесса фильтрации двухфазной жидкости на основе законов сохранения в интегральной форме[1], а в качестве системы рассматривается система LuNA.

[1] Иванов М.И., Кремер И.А., Лаевский Ю.М. Моделирование процесса фильтрации двухфазной жидкости на основе законов сохранения в интегральной форме, 2020. - 35 с.

LuNA и фрагментированное программирование

LuNA (Language for Numerical Algorithms) – система автоматического конструирования параллельных программ, разрабатываемая в ИВМиМГ СО РАН. Использует ***технологию фрагментированного программирования:***



Фрагментированная программа – это описание множества последовательных частей без побочных эффектов, которые называются *фрагментами вычислений*, связанных между собой информационно и исполняющихся по готовности данных.

Фрагменты данных Фрагменты вычислений Отношения вход-выход

Цель работы и задачи

Цель:
Разработка эффективной фрагментированной программы решения задачи фильтрации двухфазной жидкости.

Задачи:

1. выполнение анализа решения задачи фильтрации двухфазной жидкости *с точки зрения параллельной реализации*;
2. выработка требований к фрагментированной программе;
3. разработка схемы фрагментированной программы;
4. реализация полученной схемы и проведение тестирования.

Обзор работ

Рассмотрены инструменты автоматизации конструирования параллельных программ:

- LuNA
- PaRSEC
- PLASMA
- Legion
- Charm++

Решения, наработанные в этих и других системах, всех потребностей не закрывают, поэтому ***есть потребность в исследовании других частных случаев***.

Задача фильтрации

Решается матрично-векторная система: $[B^T D^{-1} B + \varepsilon C] \cdot \overline{u} = \overline{f}$

- D - симметричная трехдиагональная матрица
- B - прямоугольная разреженная матрица
- C - симметричная квадратная плотная матрица $C = (\overline{e} \cdot \overline{e}^T)$

При $\varepsilon > 0$ можно решать систему при помощи ***итерационного метода сопряженных градиентов***

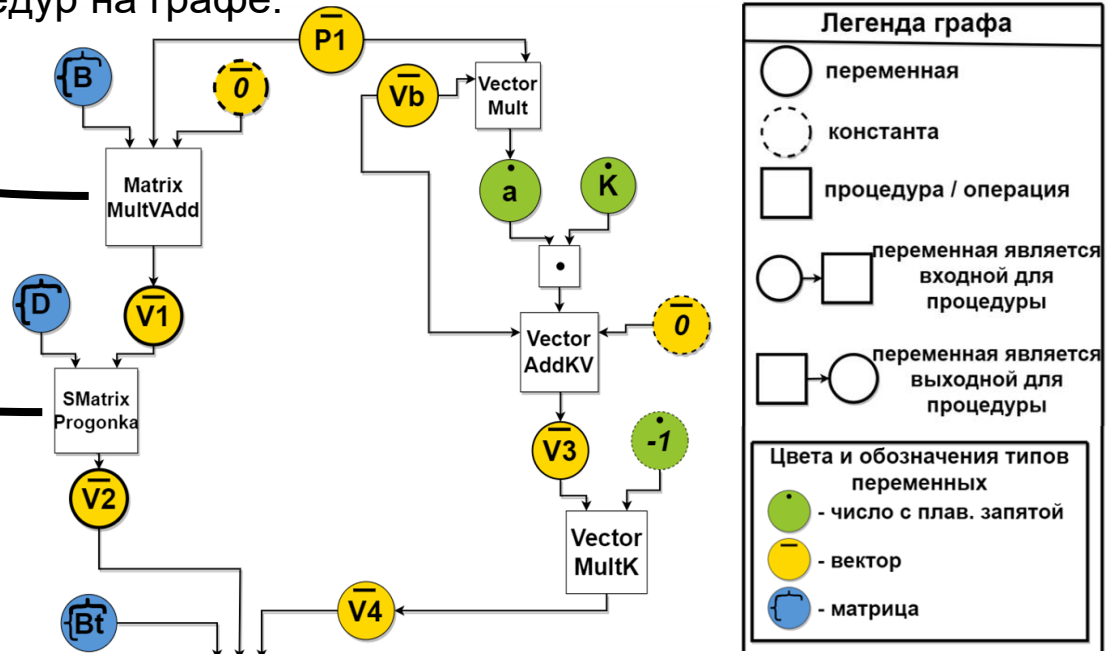
Метод сопряженных градиентов в задаче

- **Выполнен анализ** метода сопряженных градиентов в *многопоточной реализации*.
- **Разработан граф** метода сопряженных градиентов.
- **Выполнен анализ** всех процедур на графе.

Умножение матрицы на вектор

Метод прогонки

Умножение транспонированной матрицы на вектор



Требования к фрагментированной программе

Выработаны требования к фрагментированной программе.

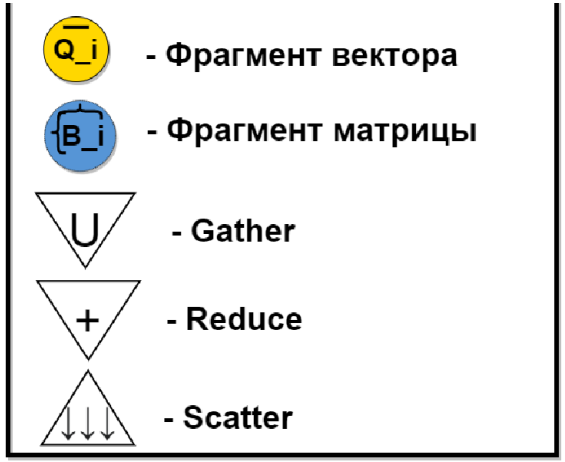
направленные на систему LuNA

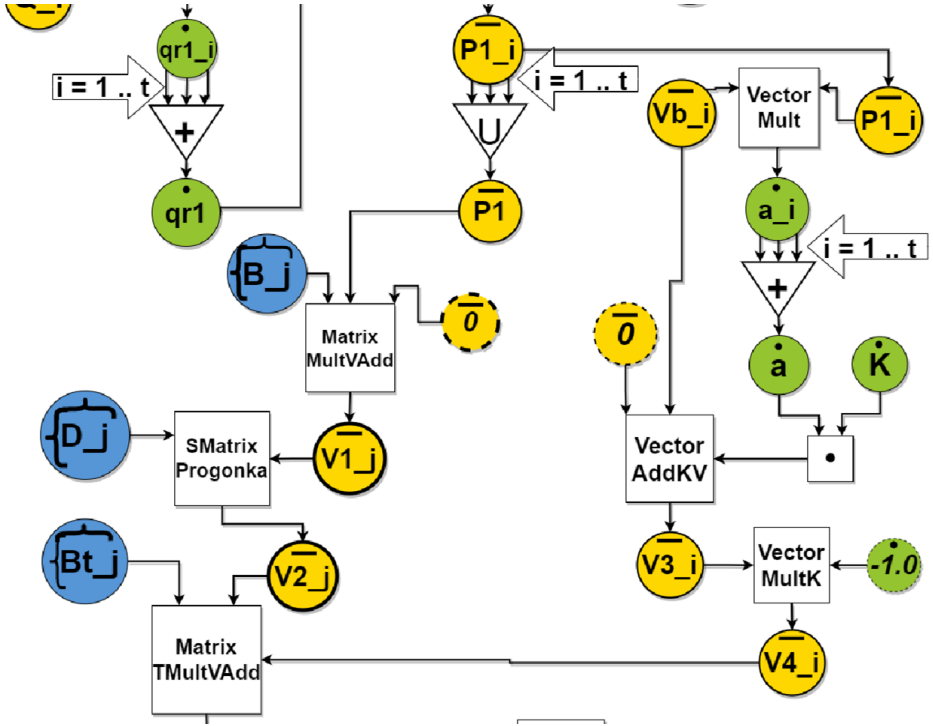
1. Процедуры, выполняющие вычисления над фрагментами данных, не должны иметь побочных эффектов.
2. Использовать примитивные типы данных языка программирования C++.

связанные с нефункциональными характеристиками

1. Программа должна решать задачу на разном количестве процессов и потоков правильно.
2. Размеры фрагментов для одного и того же вектора или матрицы должны отличаться не более чем на 1.
3. Программа не должна делать лишних коммуникаций при передаче фрагментов между процессами.
4. Полученная программа должна работать достаточно эффективно.

Схема фрагментированного метода сопряженных градиентов





Разработана эффективная фрагментированная программа с использованием MPI и OpenMP на основе анализа и возможностей параллельной реализации.

Реализация

Векторы разбиваются на фрагменты равномерно, операции выполняются независимо.

Умножение матрицы на вектор: матрица разбивается на фрагменты по строкам, входной вектор хранится полностью. В результате получается разбитый на фрагменты выходной вектор.

$$\begin{bmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,n} \\ a_{2,1} & a_{2,2} & \dots & a_{2,n} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} = \begin{bmatrix} a_{1,1}x_1 + + \dots + a_{1,n}x_n \\ a_{2,1}x_1 + \dots + a_{2,n}x_n \end{bmatrix}$$

$$\dots$$

$$\begin{bmatrix} a_{m-1,1} & a_{m-1,2} & \dots & a_{m-1,n} \\ a_{m,1} & a_{m,2} & \dots & a_{m,n} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{bmatrix} = \begin{bmatrix} f_{m-1} \\ f_m \end{bmatrix} = \begin{bmatrix} a_{m-1,1}x_1 + \dots + a_{m-1,n}x_n \\ a_{m,1}x_1 + \dots + a_{m,n}x_n \end{bmatrix}$$

Метод прогонки: матрица разбивается на несколько диагональных блоков, и метод прогонки выполняется над каждым блоком независимо. Размеры блоков соотносятся с размерами векторов в предыдущей операции.

$$\begin{bmatrix} a_1 & b_1 & \dots & 0 & 0 \\ b_1 & a_2 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & a_{m-1} & b_{m-1} \\ 0 & 0 & \dots & b_{m-1} & a_m \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \dots \\ y_{m-1} \\ y_m \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ \dots \\ f_{m-1} \\ f_m \end{bmatrix}$$

,

$$\begin{bmatrix} a_1 & b_1 \\ b_1 & a_2 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}$$

$$\dots$$

$$\begin{bmatrix} a_{m-1} & b_{m-1} \\ b_{m-1} & a_m \end{bmatrix} \begin{bmatrix} y_{m-1} \\ y_m \end{bmatrix} = \begin{bmatrix} f_{m-1} \\ f_m \end{bmatrix}$$

Реализация

Умножение транспонированной матрицы на вектор:

Матрица *разреженная*, поэтому хранится в *специальном формате (CSR)* не в транспонированном виде и это накладывает *ограничения на способ обхода* матрицы.

Матрица разбивается на фрагменты по строкам и аналогичным образом разбивается входной вектор.

$$\begin{bmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,n-1} & a_{1,n} \\ a_{2,1} & a_{2,2} & \dots & a_{2,n-1} & a_{2,n} \end{bmatrix}^T \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} f_{1,1} \\ f_{2,1} \\ \dots \\ f_{n-1,1} \\ f_{n,1} \end{bmatrix} = \begin{bmatrix} a_{1,1}x_1 + a_{2,1}x_2 \\ a_{1,2}x_1 + a_{2,2}x_2 \\ \dots \\ a_{1,n-1}x_1 + a_{2,n-1}x_2 \\ a_{1,n}x_1 + a_{2,n}x_2 \end{bmatrix}$$

$$\dots \qquad \dots \qquad \dots$$

В результате умножения и редукции результатов получается результирующий вектор.

$$\begin{bmatrix} a_{m-1,1} & a_{m-1,2} & \dots & a_{m-1,n-1} & a_{m-1,n} \\ a_{m,1} & a_{m,2} & \dots & a_{m,n-1} & a_{m,n} \end{bmatrix}^T \begin{bmatrix} x_{m-1} \\ x_m \end{bmatrix} = \begin{bmatrix} f_{1,k} \\ f_{2,k} \\ \dots \\ f_{n-1,k} \\ f_{n,k} \end{bmatrix} = \begin{bmatrix} a_{m-1,1}x_1 + a_{m,1}x_2 \\ a_{m-1,2}x_1 + a_{m,2}x_2 \\ \dots \\ a_{m-1,n-1}x_1 + a_{m,n-1}x_2 \\ a_{m-1,n}x_1 + a_{m,n}x_2 \end{bmatrix}$$

$$\begin{bmatrix} f_1 \\ f_2 \\ \dots \\ f_{n-1} \\ f_n \end{bmatrix} = \begin{bmatrix} f_{1,1} \\ f_{2,1} \\ \dots \\ f_{n-1,1} \\ f_{n,1} \end{bmatrix} + \dots + \begin{bmatrix} f_{1,k} \\ f_{2,k} \\ \dots \\ f_{n-1,k} \\ f_{n,k} \end{bmatrix}$$

Реализация

Умножение транспонированной матрицы на вектор ***с использованием потоков***:

создается несколько копий выходного вектора.

Каждый поток пишет только в определенную копию.

Потом копии суммируются.

Размеры матрицы: mxn, где m - количество потоков. n -размер выходного вектора

Шаг 1: вычисление частичных векторов:

$$\begin{bmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,n-1} & a_{1,n} \\ a_{2,1} & a_{2,2} & \dots & a_{2,n-1} & a_{2,n} \\ \dots & \dots & \dots & \dots & \dots \\ a_{m,1} & a_{m,2} & \dots & a_{m,n-1} & a_{m,n} \end{bmatrix} \begin{matrix} \leftarrow \text{сюда пишет поток 0} \\ \leftarrow \text{сюда пишет поток 1} \\ \\ \leftarrow \text{сюда пишет поток m-1} \end{matrix}$$

Шаг 2: суммирование частичных векторов:

$$\begin{matrix} \text{столбцы суммирует поток 0} & & & \text{столбцы суммирует поток m-1} \\ \downarrow & & \dots & \downarrow \\ \hline & & \dots & \hline \end{matrix}$$

$$\begin{bmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,n-1} & a_{1,n} \\ a_{2,1} & a_{2,2} & \dots & a_{2,n-1} & a_{2,n} \\ \dots & \dots & \dots & \dots & \dots \\ a_{m,1} & a_{m,2} & \dots & a_{m,n-1} & a_{m,n} \end{bmatrix}$$

$$\begin{matrix} \Downarrow & \Downarrow & \dots & \Downarrow & \Downarrow \\ [f_1 & f_2 & \dots & f_{n-1} & f_n] \end{matrix}$$

Тестирование

Размер сетки: 250x250x25

Запуски были выполнены на 2-х кластерах:

- МВС-10П [1]
- НГУ [2]

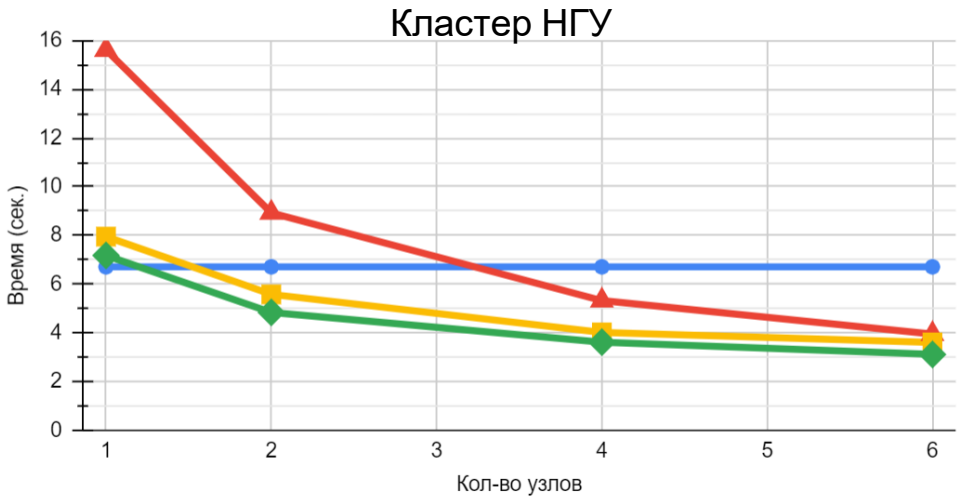
В сравнении участвовало 4 реализации:

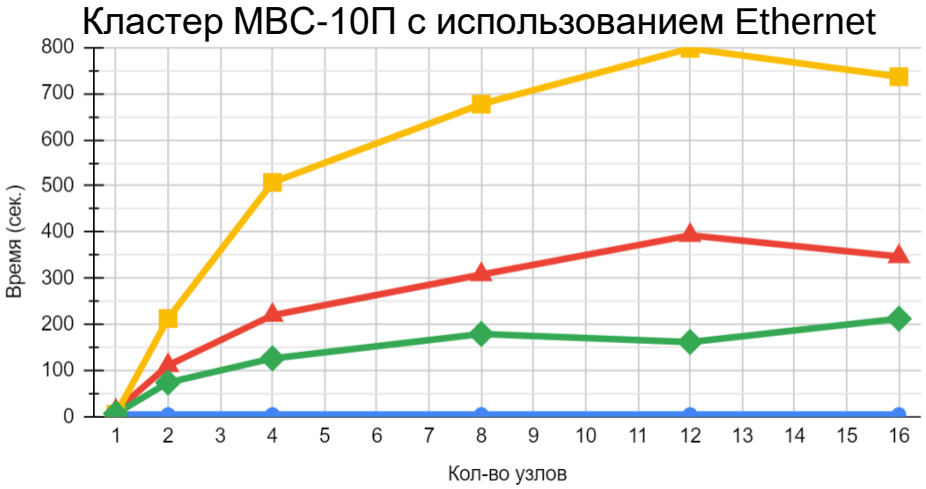
- Многопоточная
- MPI
- Фрагментированная
- MPI + OpenMP

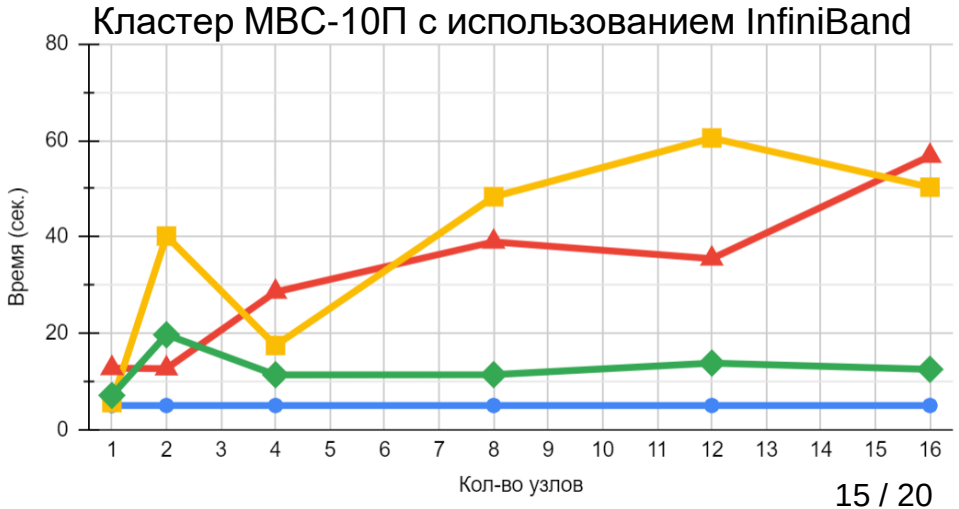
[1] https://www.jscc.ru
[2] http://nusc.nsu.ru/wiki/doku.php/doc/index

Тестирование

- Многопоточная
- MPI
- MPI + OpenMP
- Фрагментированная



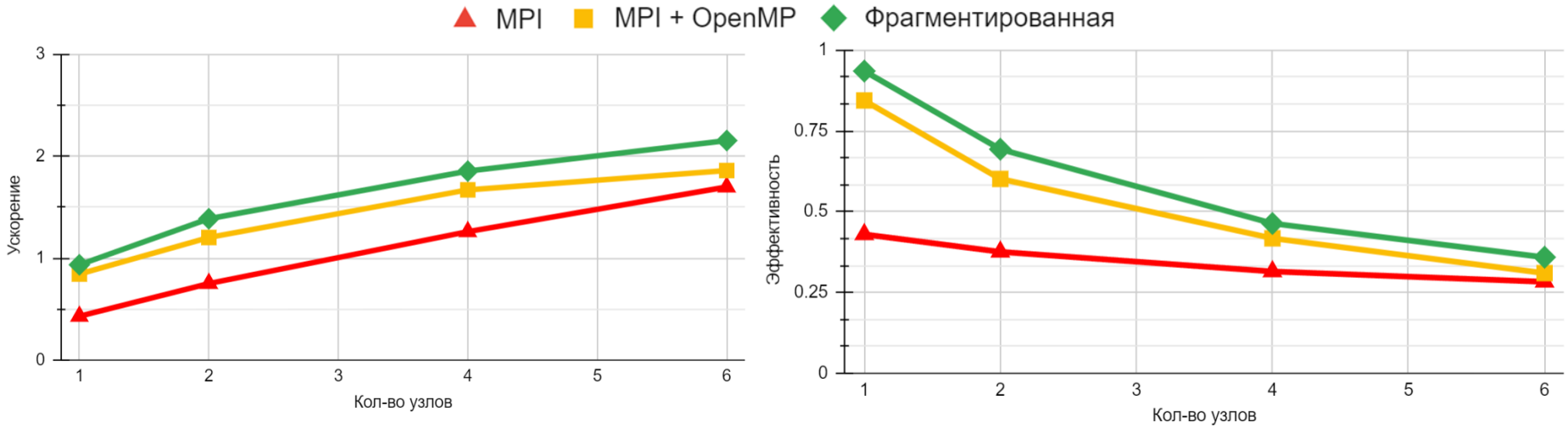




Ускорение и эффективность на кластере НГУ

Ускорение: $S_N=T_1/T_N$

Эффективность: $E_N=S_N/N$



Полученные значения эффективности *типичны* для такого класса задач и являются *удовлетворительными*. Производительность фрагментированной программы получилась сопоставимой с альтернативными реализациями.

T_1 - время работы программы на 1 узле, T_N - время работы параллельной программы на *N* узлах

Заключение

Разработана эффективная фрагментированная программа решения задачи фильтрации, в частности:

1. Проанализирована задача фильтрации двухфазной жидкости на предмет параллельной реализации.
2. Выработаны требования к фрагментированной программе.
3. Составлена схема эффективной фрагментированной программы.
4. Разработана фрагментированная программа.
5. Выполнено тестирование программы.
6. Получена удовлетворительная эффективность для такого класса задач.

На защиту выносятся

- Требования, предъявляемые к фрагментированной программе.
- Реализация фрагментированной программы.
- Экспериментальное исследование эффективности фрагментированной программы.

Дальнейшие планы:

- Улучшение системных алгоритмов в системе LuNA.
- Возможное улучшение параллелизма в алгоритмах, решающих задачу.

Апробация

1. Всероссийская летняя XXXIX молодежная Школа-конференция по параллельному программированию с международным участием

2. МНСК-2023 // Секция «Информационные технологии», подсекция «Параллельные вычисления»

Публикации

1. Всероссийская летняя XXXIX молодежная Школа-конференция по параллельному программированию с международным участием (г. Новосибирск, 4 – 15.07.2022 г.): Тез. докл. / отв. ред. д-р техн. наук, проф. В. Э. Малышкин. Новосиб. гос. унт. – Новосибирск : ИПЦ НГУ, 2023

2. Спирин В. А. Разработка эффективной фрагментированной программы решения задачи фильтрации двухфазной жидкости / В. А. Спирин // Информационные технологии. Цифровые технологии нефтегазовой индустрии : Материалы 61-й Междунар. науч. студ. конф. 17-26 апреля 2023 г. / Новосиб. гос. ун-т. — Новосибирск : ИПЦ НГУ, 2023

3. Кудрявцев А.А., Малышкин В.Э., Нуштаев Ю.Ю., Перепёлкин В.А., Спирин В.А. Эффективная фрагментированная реализация краевой задачи фильтрации двухфазной жидкости // журнал "Проблемы информатики", 2023, №2

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«НОВОСИБИРСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
(НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ, НГУ)
Факультет информационных технологий
Кафедра параллельных вычислений
09.03.01 Информатика и вычислительная техника. Программная инженерия и компьютерные науки

Разработка эффективной фрагментированной программы решения задачи фильтрации двухфазной жидкости

Выполнил: Спирин Виталий Андреевич, студент группы 19203 ФИТ НГУ

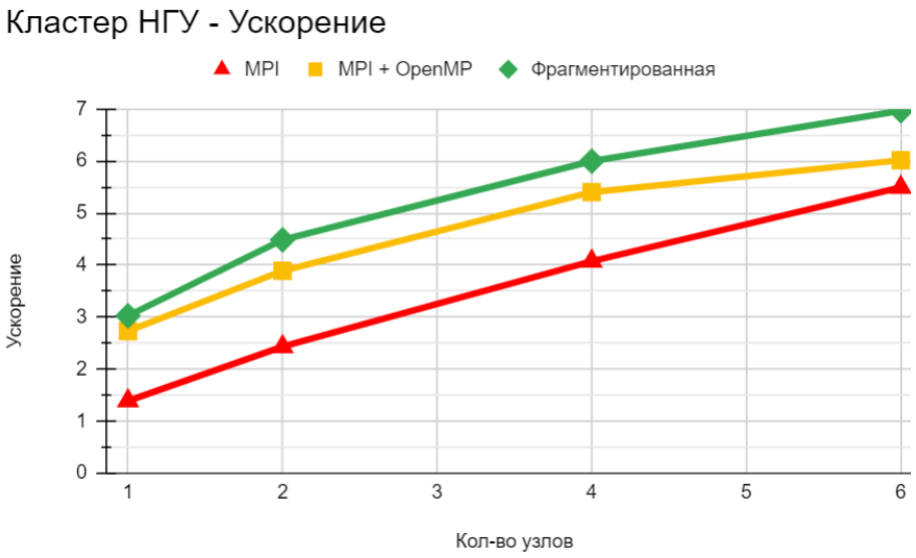
Научный руководитель: Малышкин Виктор Эммануилович, д.т.н., профессор, зав. каф. ПВ ФИТ НГУ

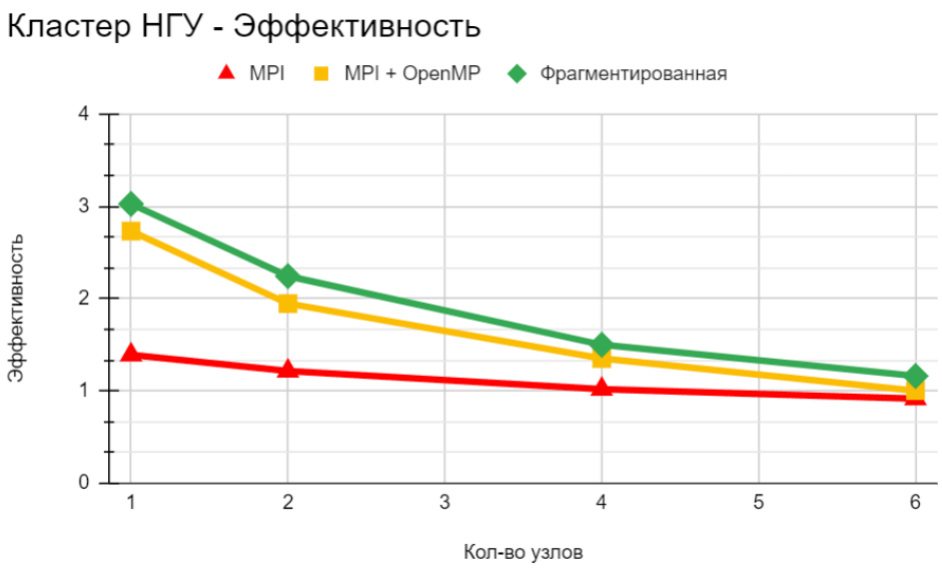
Соруководитель: Перепёлкин Владислав Александрович, ст. преп. каф. ПВ ФИТ НГУ

Рецензент: Дучков Антон Альбертович, к.ф.-м.н., зав. лаб. ИНГГ СО РАН

Новосибирск 2023

22.06.2023


Кластер НГУ - Ускорение
MPI
MPI + OpenMP
Фрагментированная
Ускорение
0
1
2
3
4
5
6
7
1
2
3
4
5
6
Кол-во узлов


Кластер НГУ - Эффективность
MPI
MPI + OpenMP
Фрагментированная
Эффективность
0
1
2
3
4
1
2
3
4
5
6
Кол-во узлов