

Разработка и реализация фрагментированных алгоритмов для работы с разреженными матрицами

Студент: Ткачёва А.А.

Руководитель: Городничев М.А.,
ассистент каф. ПВТ

Цель работы

- Работа является частью проекта по разработке библиотеки численных фрагментированных подпрограмм.

Цель работы: разработать фрагментированные алгоритмы решения системы линейных алгебраических уравнений с разреженной матрицей методом сопряженных градиентов.

- Задачи:
 - Выбрать способы фрагментации разреженных матриц.
 - Разработать и реализовать фрагментированные алгоритмы решения системы линейных алгебраических уравнений методом сопряженных градиентов.
 - Выполнить теоретическое сравнение фрагментированных алгоритмов и сравнительное тестирование фрагментированных программ с различными способами фрагментации разреженной матрицы.

Необходимые свойства параллельной программы

- Автоматическая настройка на ресурсы
 - Начальная настройка на ресурсы
 - Динамическая балансировка
 - Коммуникации на фоне вычислений
- Переносимость программы
 - Сохранение свойств программы при переходе на другую вычислительную систему

Обзор библиотек для работы с разреженными матрицами

- Библиотека PETSc
 - Начальная настройка на ресурсы
 - Коммуникации на фоне вычислений
- Библиотека PSPASES
 - Начальная настройка на ресурсы
 - Коммуникации на фоне вычислений
- Библиотека AZTEC
 - Коммуникации на фоне вычислений
- ...

Все перечисленные библиотеки поддерживают настройку на ресурсы на низком уровне и переносимость в рамках этого уровня.

Система фрагментированного программирования LuNA

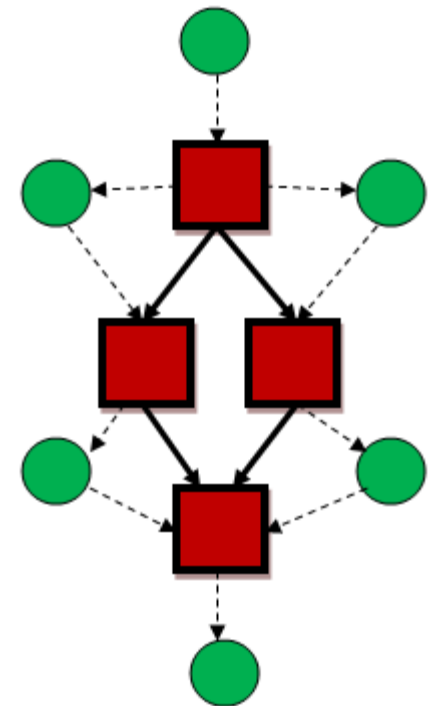
- Фрагментированный алгоритм в системе LuNA представляется в виде:

● Множество фрагментов данных (ФД)

■ Множество фрагментов вычислений (ФВ)

→ Отношение порядка на множестве фрагментов вычислений

- Фрагментированная программа:
 - Фрагментированный алгоритм
 - Частичное распределение ресурсов
 - Частичное управление



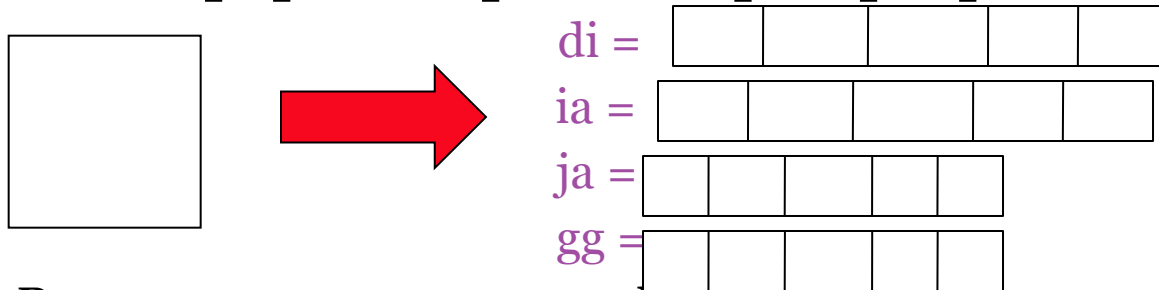
Методические рекомендации к способу фрагментации алгоритма

- ФД должны иметь ту же структуру, что и данные последовательного алгоритма, чтобы использовать для реализации ФВ готовые подпрограммы.
- Все ФД должны иметь примерно одинаковый размер, а ФВ — примерно одинаковую вычислительную сложность.
- Размер ФД и вычислительная сложность ФВ должны быть параметром алгоритма, не должны зависеть от размера задачи.

Требуется разработать способ фрагментации разреженных матриц для алгоритма метода сопряженных градиентов

Способы фрагментации разреженных матриц

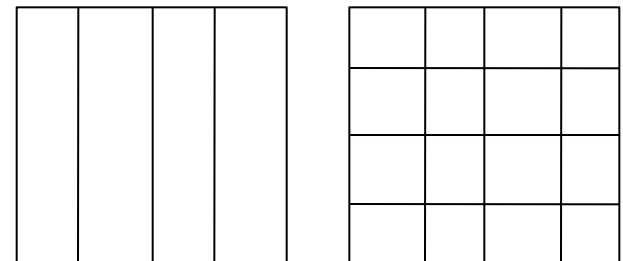
- Способ 1: массивы, представляющие матрицу в разреженном формате, равномерно разрезаются



- Способ 2: Разрезание матрицы на фрагменты и представление фрагментов в разреженном формате:

А. разрезание на полосы;

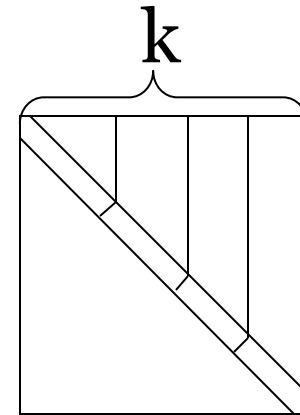
Б. разрезание на блоки.



Способы фрагментации: способ 1

- Достоинства
 - ФД одинакового размера;
 - ФВ имеют одинаковую вычислительную сложность;
 - размер ФД не зависит от размера матрицы.
- Недостатки
 - низкая степень параллелизма

Способы фрагментации: способ 2 разрезание на полосы



- Достоинства
 - более высокая степень параллелизма
- Недостатки:
 - Размеры ФД сильно различаются, зависят от размера матрицы

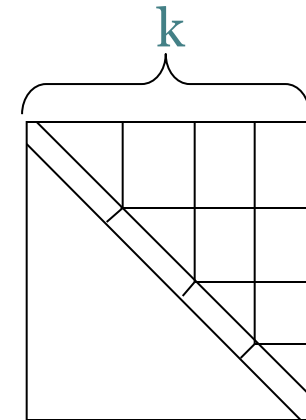
Способы фрагментации: способ 2 разрезание на блоки

- Достоинства

- Размеры ФД не зависят от размера матрицы
- ФД отличаются по размеру незначительно
- Высокая степень параллелизма

- Недостатки

- ФВ имеют разную вычислительную сложность



Сравнение способов фрагментации: по затрачиваемой памяти на хранение матрицы

Способ фрагментации	Профиль матрицы, значения	Элементы матрицы, значения
Способ 1.	$n + k + m$	$n + m$
Способ 2 полоски	$n + k + m$	$n + m$
Способ 2 блоки	$\frac{n(k+1)+k(k+1)}{2} + m$	$n + m$

n – размер матрицы;

k - степень разбиения;

m - количество внедиагональных ненулевых элементов.

Сравнение способов фрагментации: по затрачиваемой памяти на хранение матрицы

Способ фрагментации	Профиль матрицы, значения	Элементы матрицы, значения
Способ 1.	$n + k + m$	$n + m$
Способ 2 полоски	$n + k + m$	$n + m$
Способ 2 блоки	$\frac{n(k+1)+k(k+1)}{2} +$	

Способ 2 с разрезанием на блоки затрачивает больше памяти на хранение матрицы, чем остальные способы

n – размер матрицы;

k - степень разбиения;

m - количество внедиагональных ненулевых элементов.

Сравнение способов фрагментации: по количеству ФВ

- Количество ФВ для алгоритма умножения матрицы на вектор

Способ 1.	k^3
Способ 2 полосы	$\frac{k(k+1)}{2}$
Способ 2 блоки	$\frac{k(k+1)}{2}$

k – степень разбиения

Сравнение способов фрагментации: по количеству ФВ

- Количество ФВ для алгоритма умножения матрицы на вектор

Способ 1.	k^3
Способ 2 полоски	$k(k + 1)$
Способ 2 блоки	

Для способа 1. требуется больше всех количество ФВ.

Для способа 2 с разрезанием на блоки достигается более высокая степень параллелизма, при таком же количестве ФВ, как и для способа 2 с разрезанием на полоски

k – степень разби

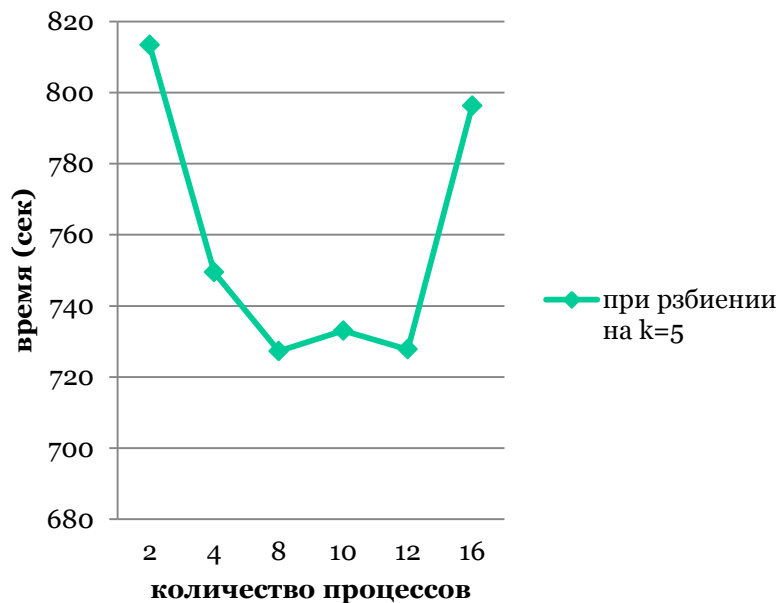
Реализация

В системе LuNA были реализованы фрагментированные программы метода сопряженных градиентов с использованием способов фрагментации 2 с разрезанием на полосы и на блоки

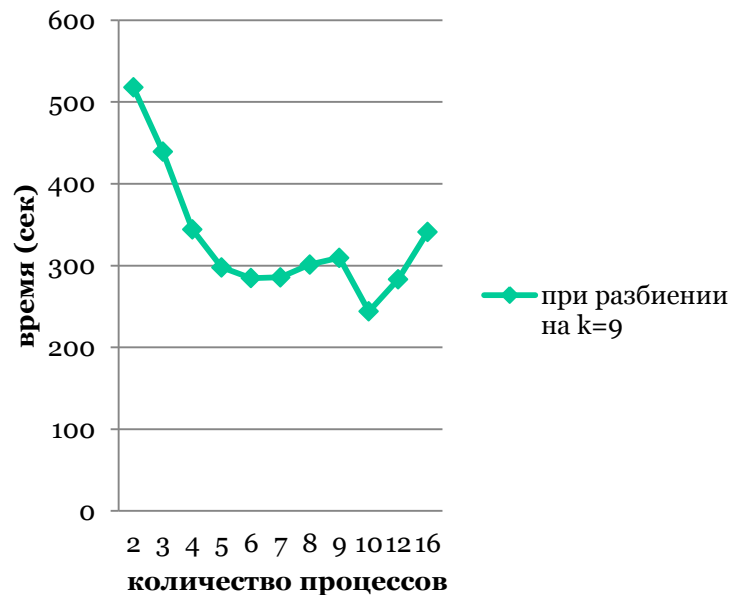
Тестирование: сравнение способов фрагментации

- Тестирование проводилось на матрице размером 4545×4545 на кластере МВС-100К

Фрагментация на полосы



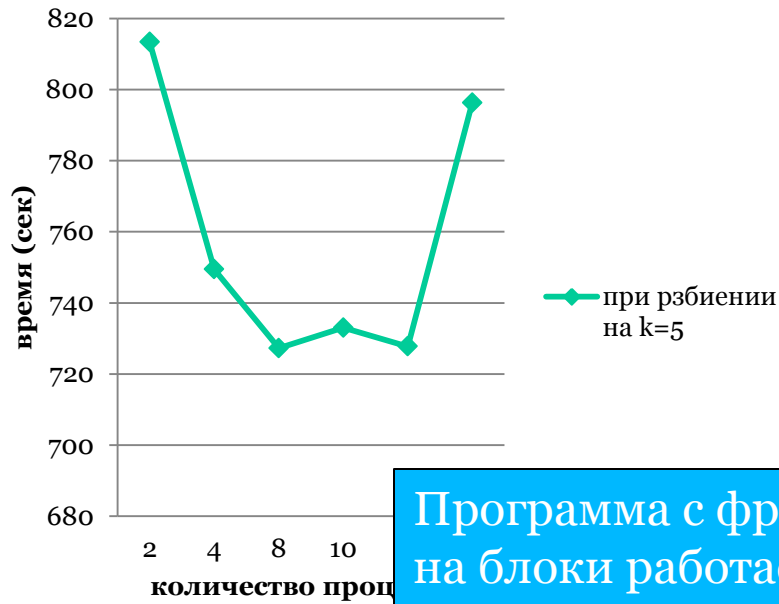
Фрагментация на блоки



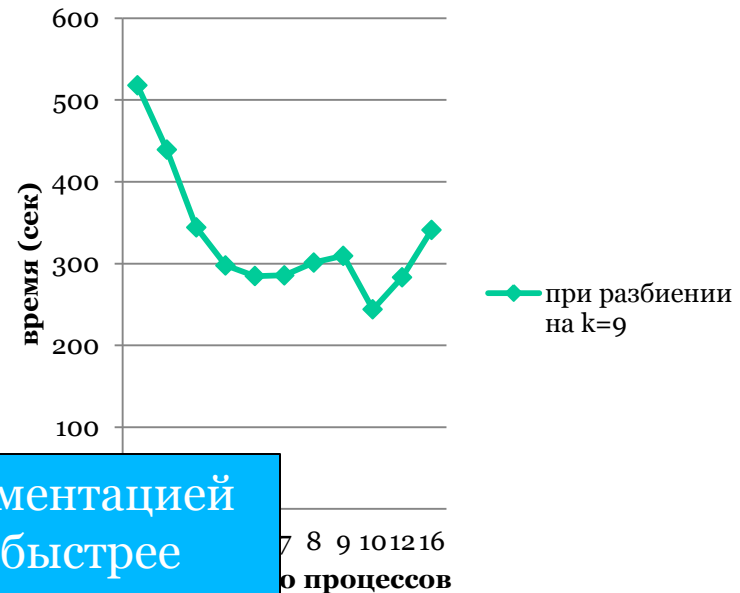
Тестирование: сравнение способов фрагментации

- Тестирование проводилось на матрице размером 4545×4545 на кластере МВС-100К

Фрагментация на полосы



Фрагментация на блоки



Программа с фрагментацией на блоки работает быстрее

Выводы

- Несмотря на то, что при фрагментации на блоки требуется больше памяти на хранение матрицы, его применение позволяет достигнуть большей степени параллелизма для фрагментированного алгоритма и реализованная фрагментированная программа работает быстрее, чем для других способов фрагментации.
- Для разреженной матрицы наиболее эффективна фрагментация на блоки.

Заключение

- Выполнен обзор и анализ параллельных библиотек для работы с разреженными матрицами
- Рассмотрены различные способы фрагментации разреженных матриц.
- Разработаны фрагментированные алгоритмы, реализующие МСГ для различных способов фрагментации разреженной матрицы.
- Алгоритмы реализованы в системе LuNA.
- Выполнено сравнительное тестирование фрагментированных программ для различных способов фрагментации разреженной матрицы.

Спасибо за внимание!