Министерство образования Республики Беларусь

Учреждение образования
“Белорусский государственный университет
информатики и радиоэлектроники”

Факультет информационных технологий и управления

Кафедра интеллектуальных информационных технологий

Отчёт по лабораторной работе №2 по курсу «МРЗвИС» на тему: «Реализация модели решения задачи на ОКМД архитектуре»

Выполнил
студент группы
821703:
Лихач Р. А.

Проверили:
Орлова А.С.
Крачковский Д.Я.

Минск 2020

Цель:

Реализовать и исследовать модель решения на ОКМД архитектуре задачи вычисления матрицы значений.

Постановка задачи:

Дано: сгенерированные матрицы **A**, **B**, **E** и **G** заданных размерностей **pxm**, **mxq**, **1xm** и **pxq** соответственно со значениями в диапазоне [-1;1].

$$c_{ij} = \tilde{\bigwedge_k} f_{ijk} * (3 * g_{ij} - 2) * g_{ij} + (\tilde{\bigvee_k} d_{ijk} + (4 * (\tilde{\bigwedge_k} f_{ijk} \tilde{\circ} \tilde{\bigvee_k} d_{ijk}) - 3 * \tilde{\bigvee_k} d_{ijk}) * g_{ij}) * (1 - g_{ij})$$
$$f_{ijk} = (a_{ik} \tilde{\rightarrow} b_{kj}) * (2 * e_k - 1) * e_k + (b_{kj} \tilde{\rightarrow} a_{ik}) * (1 + (4 * (a_{ik} \tilde{\rightarrow} b_{kj}) - 2) * e_k) * (1 - e_k)$$
$$d_{ijk} = a_{ik} \tilde{\wedge} b_{kj}$$

Вариант индивидуального задания (вариант 5):

$$\tilde{\bigwedge_k} f_{ijk} = \prod_k f_{ijk}$$
$$\tilde{\bigvee_k} d_{ijk} = 1 - \prod_k (1 - d_{ijk})$$
$$\tilde{\bigwedge_k} f_{ijk} \tilde{\circ} \tilde{\bigvee_k} d_{ijk} = max(\{\tilde{\bigwedge_k} f_{ijk} + \tilde{\bigvee_k} d_{ijk} - 1\} \cup \{0\})$$
$$a_{ik} \tilde{\rightarrow} b_{kj} = a_{ik} * (1 - b_{kj}) + 1$$
$$b_{kj} \tilde{\rightarrow} a_{ik} = b_{kj} * (1 - a_{ik}) + 1$$
$$a_{ik} \tilde{\wedge} b_{kj} = a_{ik} * b_{kj}$$

Получить: **C** – матрицу значений соответствующей размерности **pxq**; в случае необходимости доопределить всеобщности(∀) или существования(∃) условие исходной задачи кванторами самостоятельно.

Исходные данные:

1) **p**, **m**, **q** - размерность матриц;
2) **n** – количество процессорных элементов в системе;
3) **ti**– время выполнения i операции над элементами матриц;
4) матрицы **A**, **B**, **E**, **G**, заполненные случайными вещественными числами в диапазоне [-1;1].

Описание модели:

Была реализована модель решения на ОКМД архитектуре задачи вычисления матрицы значений. Возможность самостоятельно устанавливать все параметры, необходимые для работы модели, позволяет детально исследовать разработанную модель, установить зависимости между вышеуказанными параметрами.

- **T1** – время выполнения программы на одном процессорном элементе. Данный параметр вычисляется следующим образом: подсчитывается количество вызовов той или иной операции, а затем полученное значение умножается на время данной операции. Данное действие повторяется для всех операций, в итоге все значения суммируются.
- **Tn** – время выполнения программы на n-количестве процессорных элементов.

Параметр вычисляется схожим путём, что и **T1**: осуществляется поиск операций, которые можно считать на различных процессорах. Для подсчета времени на выполнение такой операции находится количество вызовов данной операции и делится на количество процессорных элементов.

- **Ky** – коэффициент ускорения равен **T1**/**Tn**.
- **e** – эффективность равна **Ky**/**n**.
- **D** - коэффициент расхождения программы, **D**=**Ls**/**Lcp**. Где, **Ls** - суммарная длина программы и равна **Tn**. **Lcp** - средняя длина программы. Вычисляется путем подсчета количества вызовов операций на различных ветвях выполнения программы. Имея, количества вызовов операций, выполняющихся на ветвях программы, и их время выполнения, считаем данную величину.

Результаты счёта и времена их получения:

```
"D:\╠╨╠т╚╘\╤рср 2\2.exe"
m = 1
p = 2
q = 3
n = 4
A:
-0.0654
-0.6435

B:
  0.978 -0.6154  0.6825

E:
-0.2447

G:
-0.6669  -0.002 -0.6386
 0.1751  -0.147 -0.0072

C:
   -0.323482  -0.00158298    -0.303182
    0.308045    -0.681666   -0.0173641

Parameters:
T1 = 240
Tn = 84
Ky = 2.85714
e = 0.714286
Lsum = 84
Lavg = 35
D = 2.4
```

Построение графиков:

Обозначения:

- Ky(n, r) – коэффициент ускорения;

- e(n, r) – эффективность;
- D(n, r)– коэффициент расхождения программы;
- n – количество процессорных элементов в системе (совпадает с количеством этапов конвейера);
- r – ранг задачи (количество объектов, которые в процессе решения задачи могли бы обрабатываться параллельно);

Графики строятся на одном наборе сгенерированных данных, постепенно уменьшая размеры матриц, в масштабе, отражающем характерные особенности соответствующих зависимостей.

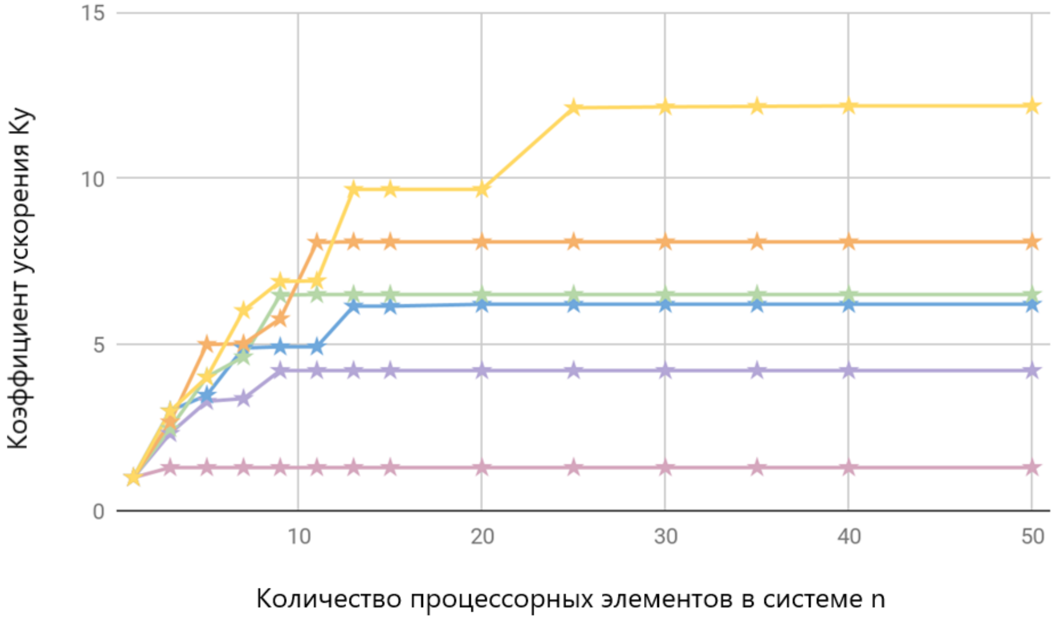

График 1. График зависимости коэффициента ускорения Ky от количества элементов n

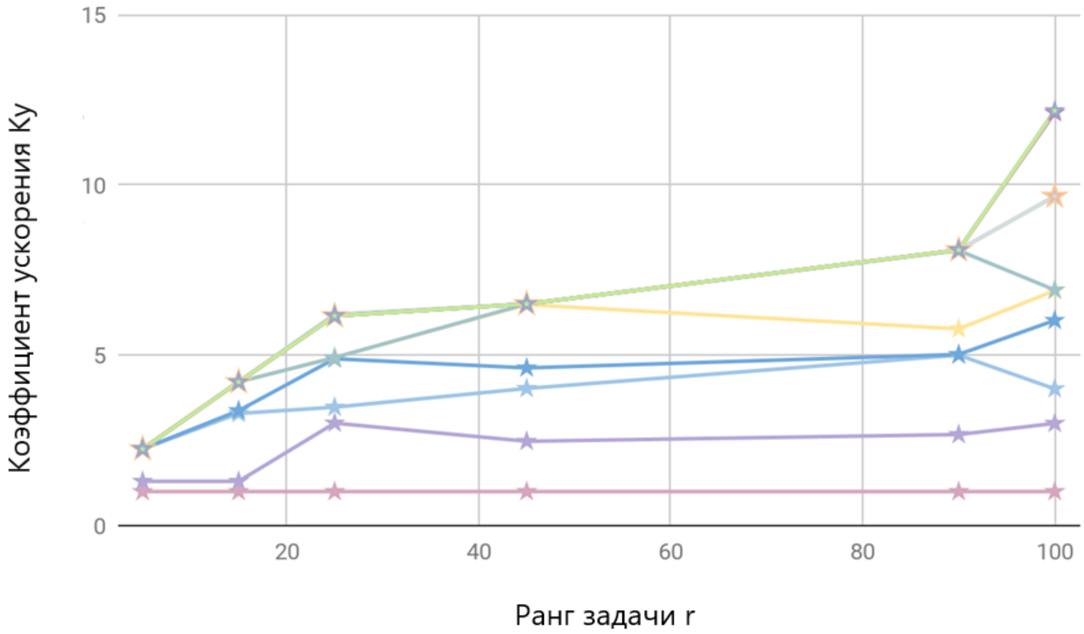

График 2. График зависимости коэффициента ускорения Ky от ранга задачи r

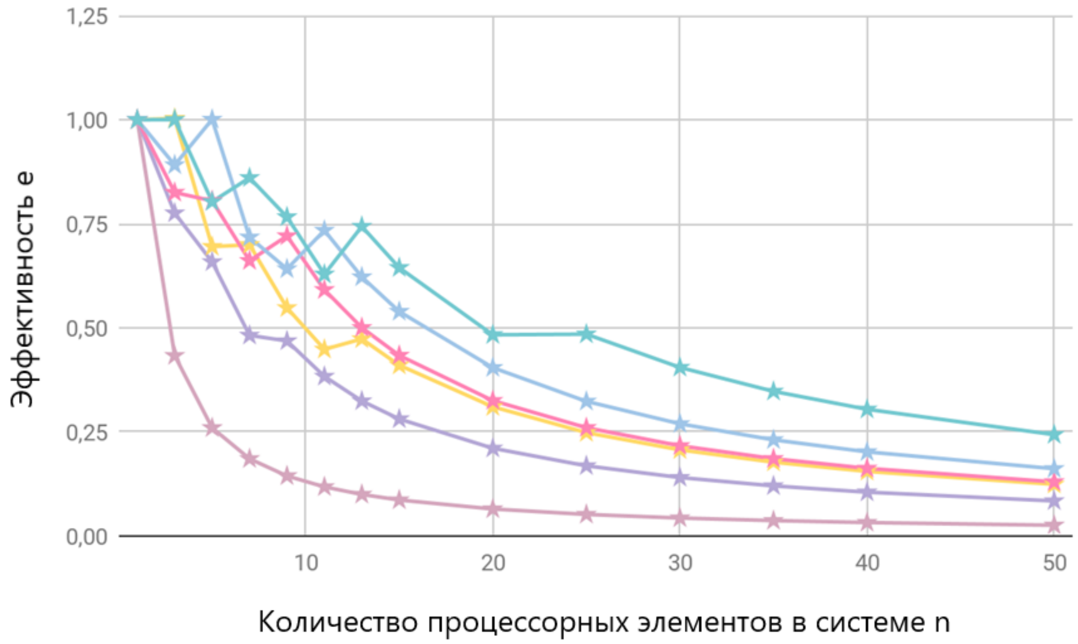

График 3. График зависимости эффективности e от количества элементов n

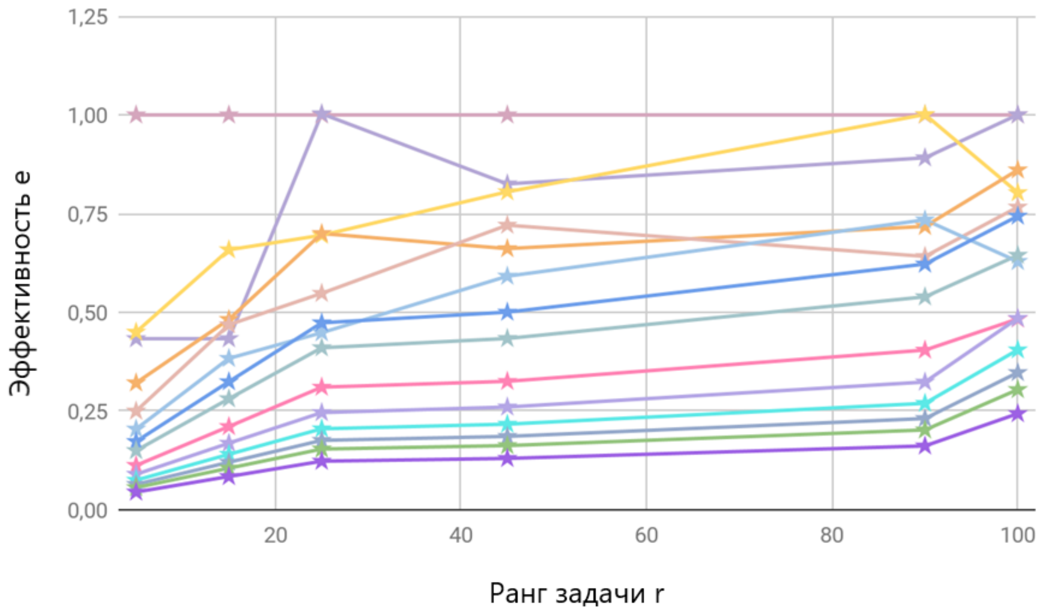

График 4. График зависимости эффективности e от ранга задачи r

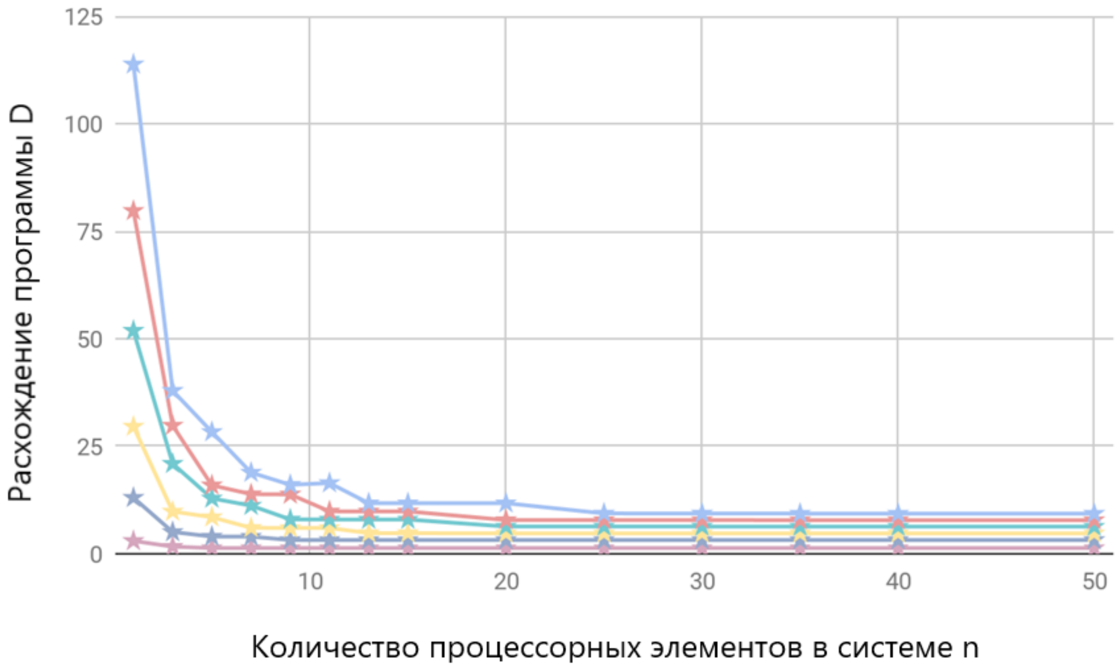

График 5. График зависимости коэффициента расхождения программы D от количества элементов n

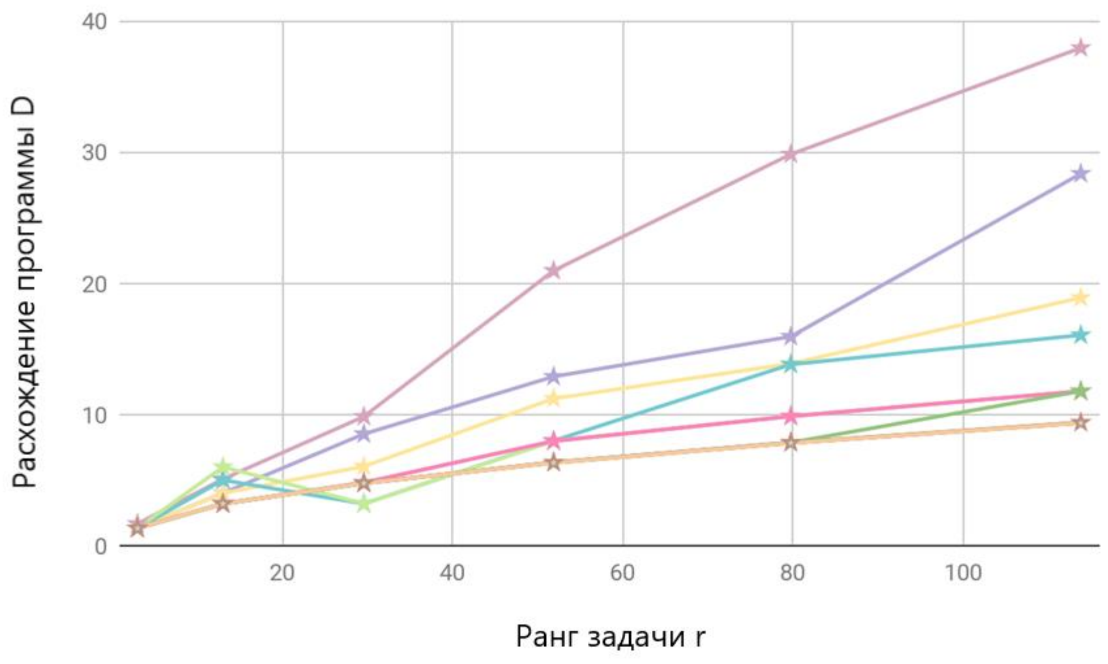

График 6. График зависимости коэффициента расхождения программы D от ранга задачи r

Ответы на вопросы:

1. Проверить, что модель создана верно: программа работает правильно;

Исходные данные:

```
m = 1
p = 2
q = 3
n = 4
A:
-0.0654
-0.6435

B:
  0.978 -0.6154  0.6825

E:
-0.2447

G:
-0.6669  -0.002 -0.6386
 0.1751  -0.147 -0.0072
```

Полученные данные:

```
C:
   -0.323482  -0.00158298    -0.303182
    0.308045    -0.681666   -0.0173641
```

Программа работает правильно.

2. Объяснить на графиках точки перегиба и ассимптоты:

*График 1 (зависимость коэффициента ускорения **Ky** от количества элементов **n**):*
Асимптотой графика, исходя из значений графика, является прямая, параллельная оси абсцисс, то есть прямая, заданная при n=r. Точки перегиба появляются тогда, когда ширина векторного параллелизма становится кратной числу процессорных элементов, при достижении этого значения коэффициент ускорения перестает расти.

*График 2 (зависимость коэффициента ускорения **Ky** от ранга задачи **r**):*
Асимптотой является прямая Ky=n, такого значения она достигает в точках , где ширина векторного параллелизма становится кратной числу процессорных элементов. При фиксированном значении процессорных элементов и при устремлении ранга задачи к бесконечности, ОКМД архитектура будет работать быстрее не более, чем в n раз по сравнению с последовательной системой.

*График 3 (зависимость эффективности **e** от количества элементов **n**):*
Прямая e=0 будет являться асимптотой. Так как задача с фиксированным рангом содержит фиксированное количество операций, которые необходимо выполнить, а эффективность показывает долю работы одного процессорного элемента, то при большом количестве процессорных элементов эффективность стремится к 0

*График 4 (зависимость эффективности **e** от ранга задачи **r**):*
Прямая e=1 будет являться асимптотой, а точками перегиба – точки, где ширина векторного параллелизма становится кратной числу процессорных элементов.

*График 5 (зависимость коэффициента расхождения программы **D** от количества элементов **n**):*
При увеличении количества элементов, значение расхождения программы стремится к 1.

*График 6 (зависимость коэффициента расхождения программы **D** от ранга задачи **r**):*
При увеличении ранга задачи, значение расхождения программы увеличивается.

3. Спрогнозировать как изменится вид графиков при изменении параметров модели; если модель позволяет, то проверить на ней правильность ответа;

*1) Для зависимости коэффициента ускорения **Ky** от количества элементов **n**:*
При увеличении количества пар элементов, возрастает значение коэффициента ускорения, до момента пока ширина векторного параллелизма не становится равной числу процессорных элементов. Далее при увеличении, коэффициент ускорения остается постоянным.

*2) Для зависимости коэффициента ускорения **Ky** от ранга задачи **r**:*
При увеличении количества процессорных элементов, возрастет значение коэффициента ускорения. Пиковые значения зафиксированы в точках, где ширина векторного параллелизма становится равной числу процессорных элементов, в этих точках **Ky**=**n**.

*3) Для зависимости эффективности **e** от количества элементов **n**:*
При увеличении количества процессорных элементов, снижается значение эффективности.

*4) Для зависимости эффективности **e** от ранга задачи **r**:*
При увеличении ранга, возрастает значение эффективности.
Пиковые значения зафиксированы в точках, где ширина векторного параллелизма становится кратной числу процессорных элементов.

*5) Для зависимости коэффициента расхождения программы **D** от количества элементов **n**:*
При увеличении количества процессорных элементов, возрастает коэффициент расхождения программы.

*6) Для зависимости коэффициента расхождения программы **D** от ранга задачи **r**:*
При увеличении ранга задачи, снижается значение коэффициента расхождения программы.

Выводы:

В результате выполнения лабораторной работы была реализована и исследована ОКМД модель для решения задач вычисления матрицы значений. Реализованная модель была проверена на работоспособность и правильность получаемых результатов. Данная модель позволяет ускорить процесс вычисления результата для числовых векторов, по сравнению с последовательной системой. Были исследованы характеристики конвейерной архитектуры: коэффициент ускорения, коэффициент расхождения программы и эффективность.