Учреждение образования
«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ»

Кафедра интеллектуальных информационных технологий

Отчет по лабораторной работе №2
по курсу
«Модели решения задач в интеллектуальных системах»

Вариант 13

Выполнил
студент группы 121702: Голушко Д.С.

Проверил: Ивашенко В. П.

МИНСК 2024

Тема: реализация модели решения задачи на ОКМД архитектуре

Цель: реализовать и исследовать модель решения на ОКМД архитектуре задачи вычисления матрицы значений.

Состав группы:

1. Голушко Даниил Сергеевич (нахождение матрицы значений, написание отчёта)
2. Нагла Никита Юрьевич (нахождение ранга задачи, коэффициента ускорения, эффективность, реализация Frontend)

Дано: сгенерированные матрицы $A,\ B,\ C,\ G$ заданных размерностей $p\times m$, $m\times q$, $1\times m$, $p\times q$ соответственно со значениями в диапазоне $[-1; 1]$.

$$c_{ij}=\underset{k}{\widetilde{\wedge}} f_{ijk}*(3*g_{ij}-2)*g_{ij}+(\underset{k}{\widetilde{\vee}} d_{ijk}+(4*(\underset{k}{\widetilde{\wedge}} f_{ijk}\,\widetilde{\circ}\,\underset{k}{\widetilde{\vee}} d_{ijk})-3*\underset{k}{\widetilde{\vee}} d_{ijk})*g_{ij})*(1-g_{ij})$$

$$f_{ijk}=(a_{ik}\widetilde{\to}b_{kj})*(2*e_k-1)*e_k+(b_{kj}\widetilde{\to}a_{ik})*(1+(4*(a_{ik}\widetilde{\to}b_{kj})-2)*e_k)*(1-e_k)$$

$$d_{ijk}=a_{ik}\widetilde{\wedge}b_{kj}$$

x /1\ y = min({x}υ{y})

x /2\ y = x * y

x /3\ y = max({x + y - 1}υ {0})

<u>*Вариант индивидуального задания*: 13</u>

<u>Описание модели:</u>

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.

1. T_1 — 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.
2. T_n — время выполнения программы на n-количестве процессорных элементов. Параметр вычисляется схожим путём, что и T_1: осуществляется

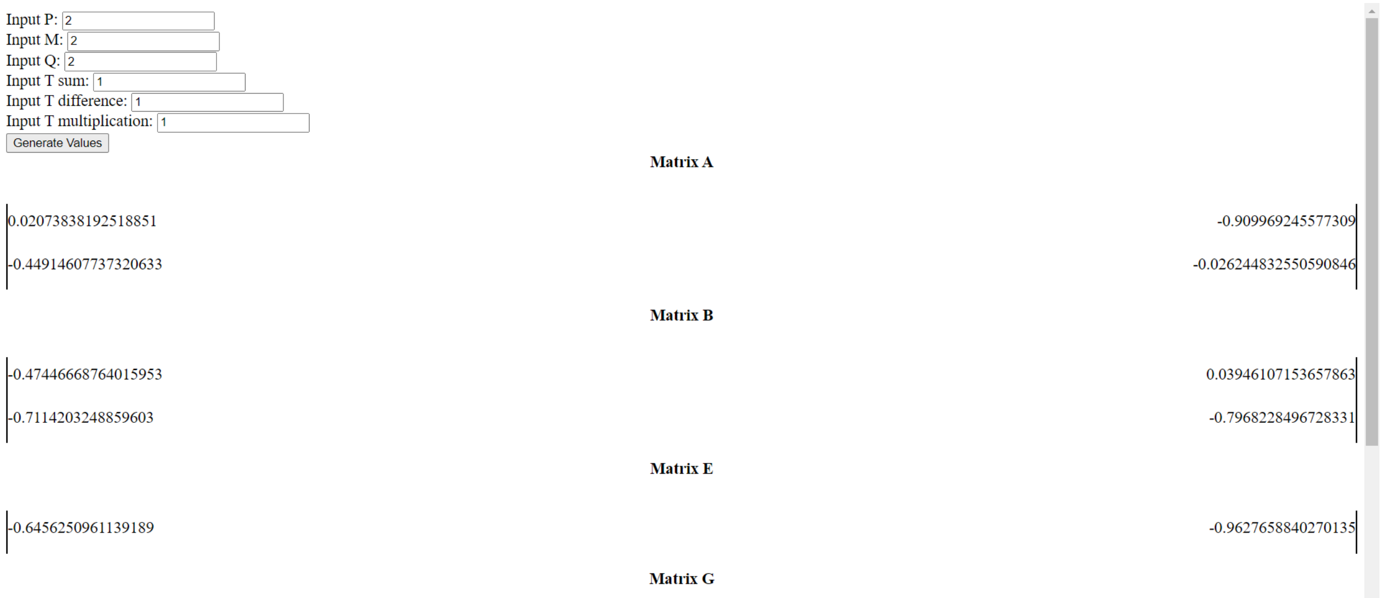
поиск операций, которые можно считать на различных процессорах. Для подсчета времени на выполнение такой операции находится количество вызовов данной операции и делится на количество процессорных элементов.

3. K_y — коэффициент ускорения равен $\frac{T_1}{T_n}$.
4. e — эффективность равна $\frac{K_y}{n}$.
5. D — коэффициент расхождения программы, $\frac{L_\Sigma}{L_{ср}}$. Где, L_Σ — суммарная длина программы и равна T_n. $L_{ср}$ — 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.

Исходные данные:

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

Результат выполнения программы:



Matrix G

-0.7812132336216724	-0.23204458268647565
-0.2544297863366882	-0.6962833146002083

Calculate

Matrix C

10.951981611586143	0.7609830771916128
3.197026476135124	16.353495077727384

T: 204 Tn: 32 a: 6.375 e: 0.796875 d: 102

Графики:

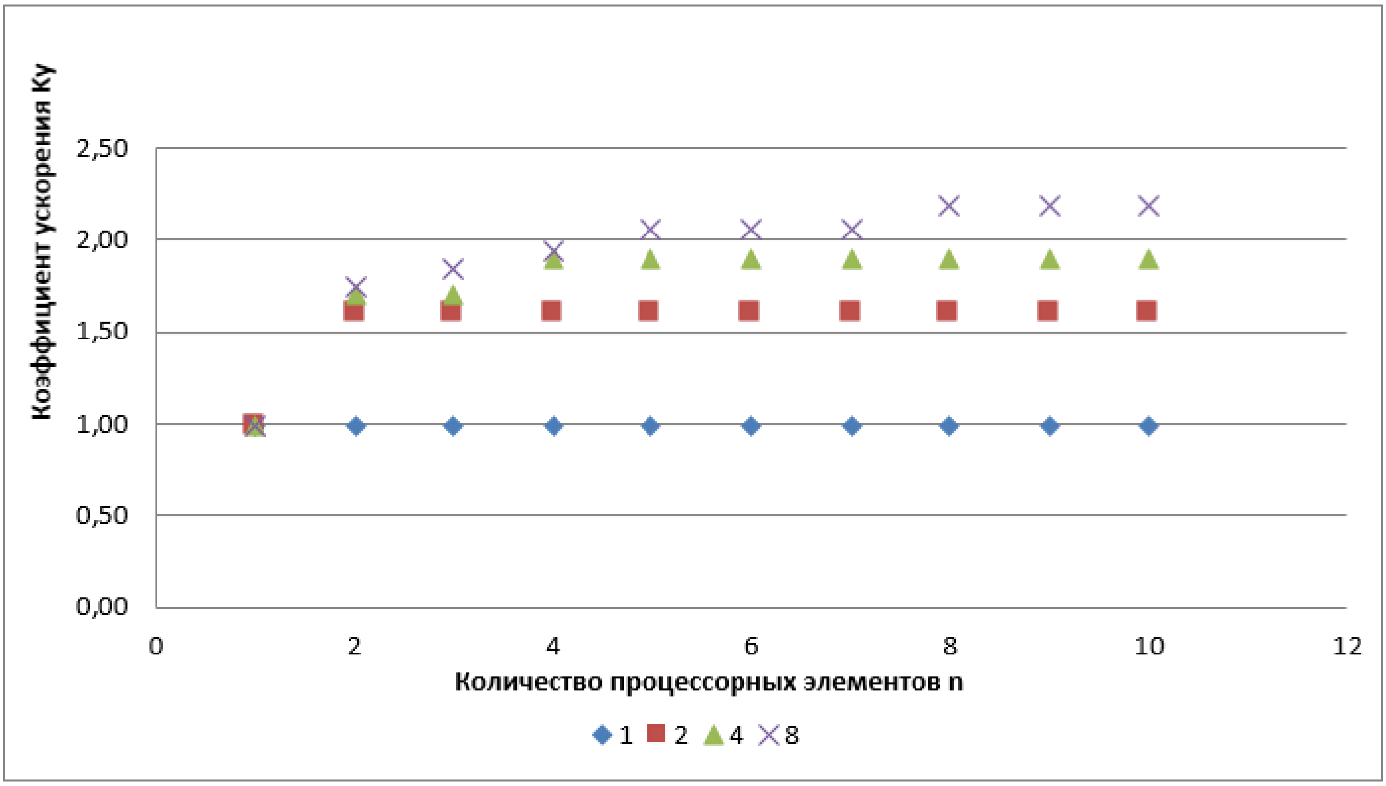


График 1. График зависимости коэффициента ускорения K_y от количества элементов n

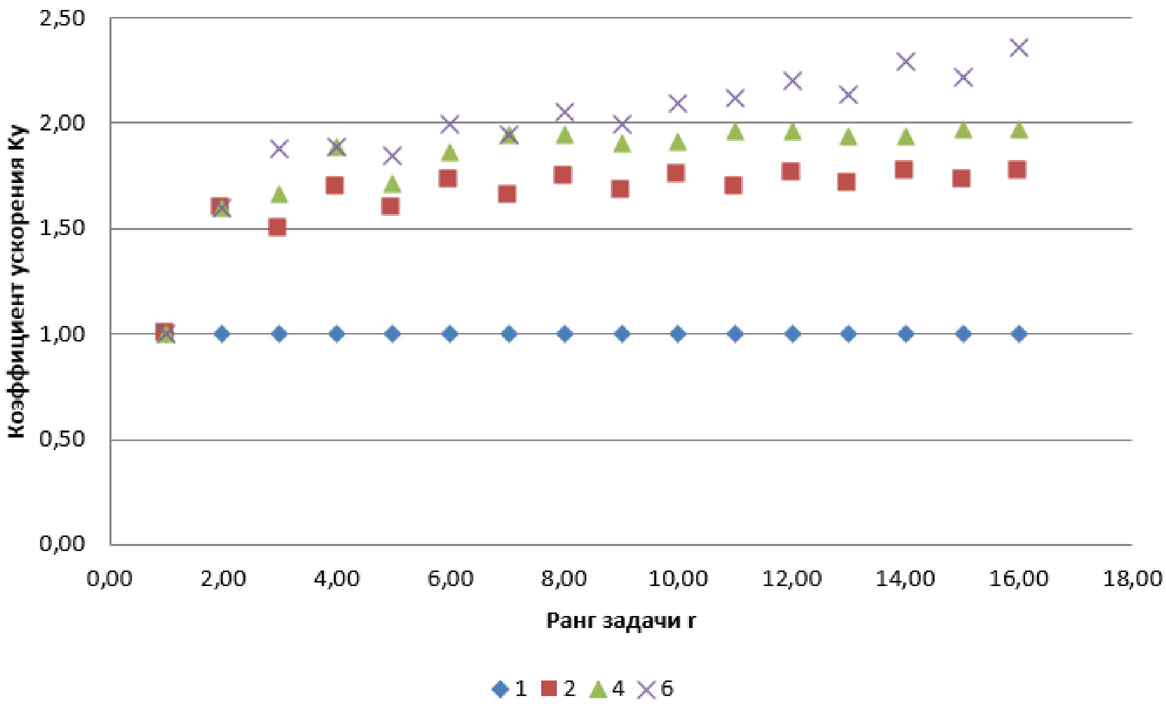


График 2. График зависимости коэффициента ускорения $K_у$ от ранга задачи r

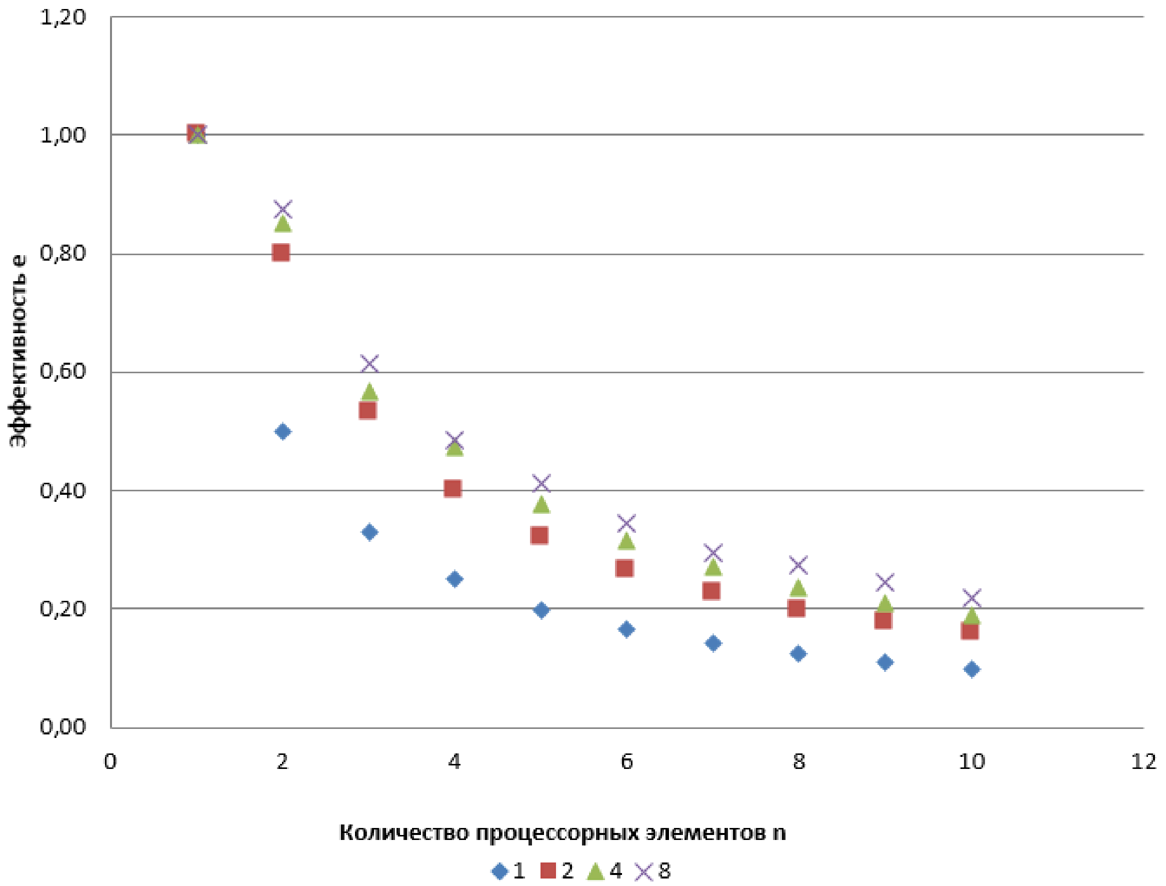


График 3. График зависимости эффективности е от количества элементов n

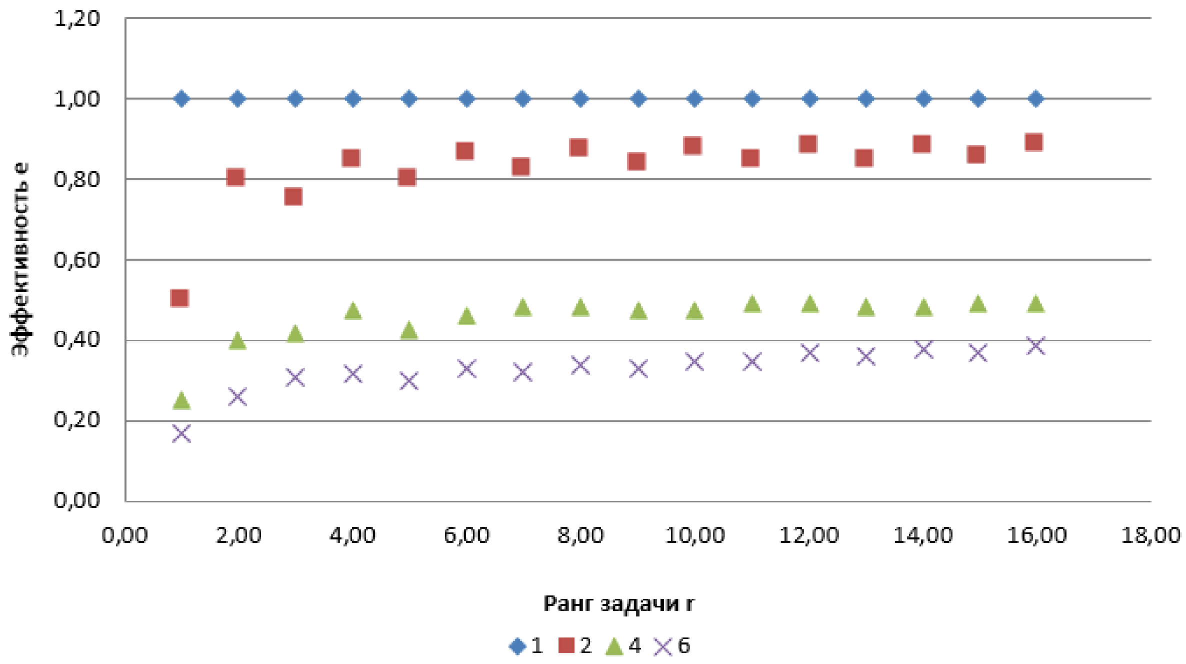


График 4. График зависимости эффективности е от ранга задачи r

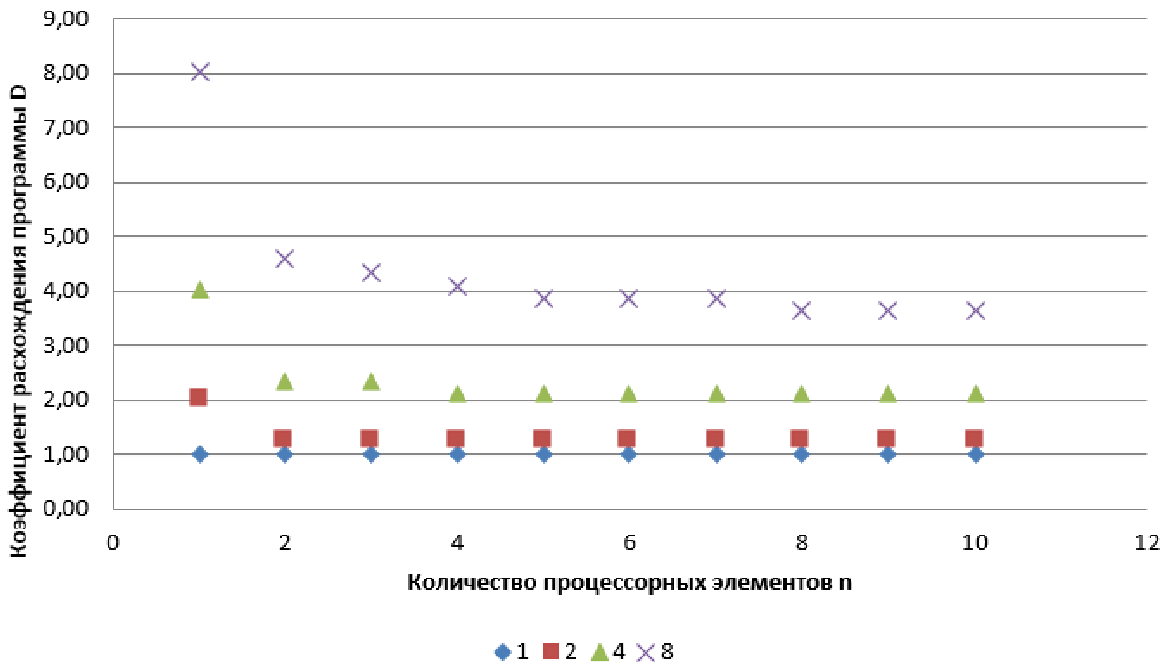


График 5. График зависимости коэффициента расхождения программы D от количества элементов n

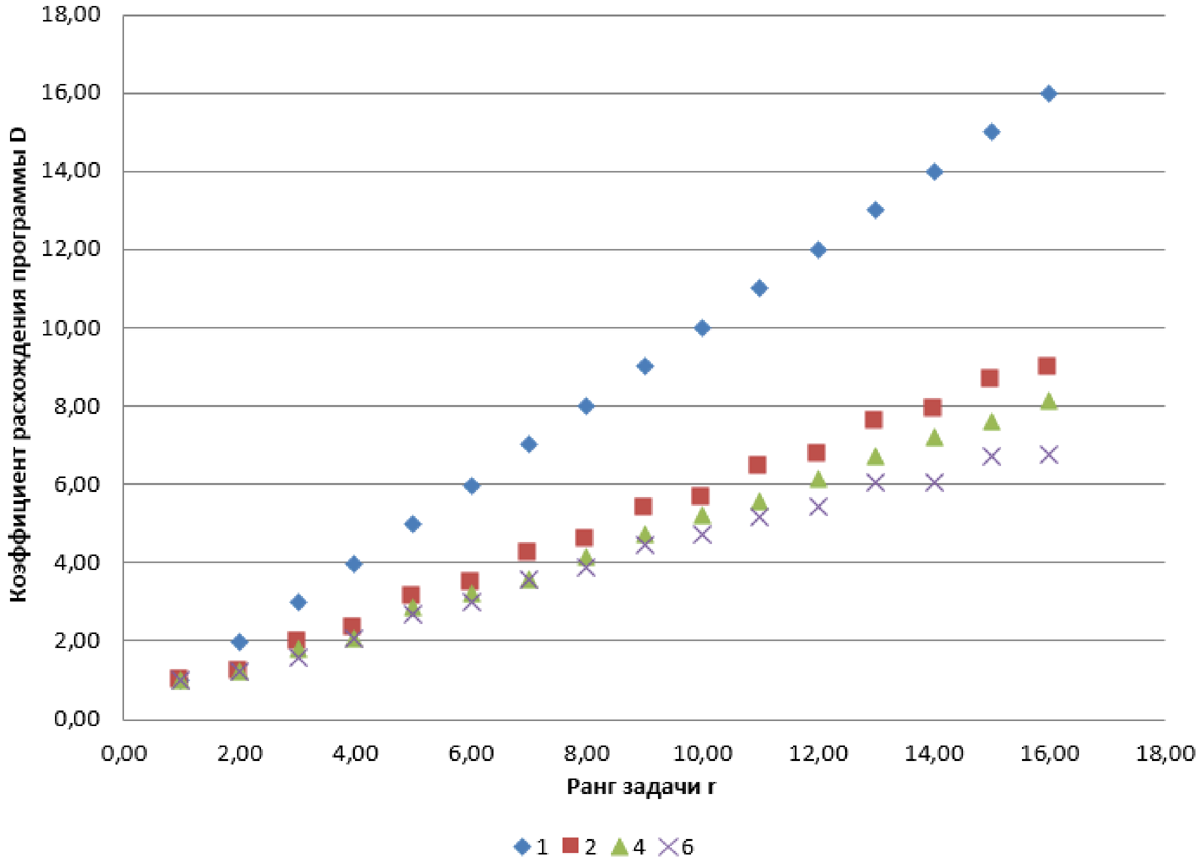


График 6. График зависимости коэффициента расхождения программы D от ранга задачи r

Вопросы:

1. *Проверить правильность работы программы:*

   p = 2, m = 2, q = 2, n = 8, T sum = 1, T difference = 1, T multiplication = 1

   Программа работает верно.

2. *Объяснить на графиках точки перегиба и асимптоты:*

   Для графика зависимости коэффициента ускорения (K_y) 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.

Для графика зависимости коэффициента ускорения (K_y) от ранга задачи (r):

Асимптотой является прямая $K_y = n$, 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.

Для графика зависимости 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

Для графика зависимости 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.

Для графика зависимости коэффициента расхождения программы (D) от количества элементов (n):

При увеличении количества элементов, значение расхождения программы стремится к 1.

Для графика зависимости 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:

   Зависимость коэффициента ускорения (K_y) 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.

Зависимость коэффициента ускорения (K_y) 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 $K_y = n$.

Зависимость эффективности (e) от количества элементов (n):

При увеличении количества процессорных элементов, снижается значение эффективности

Зависимость 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.

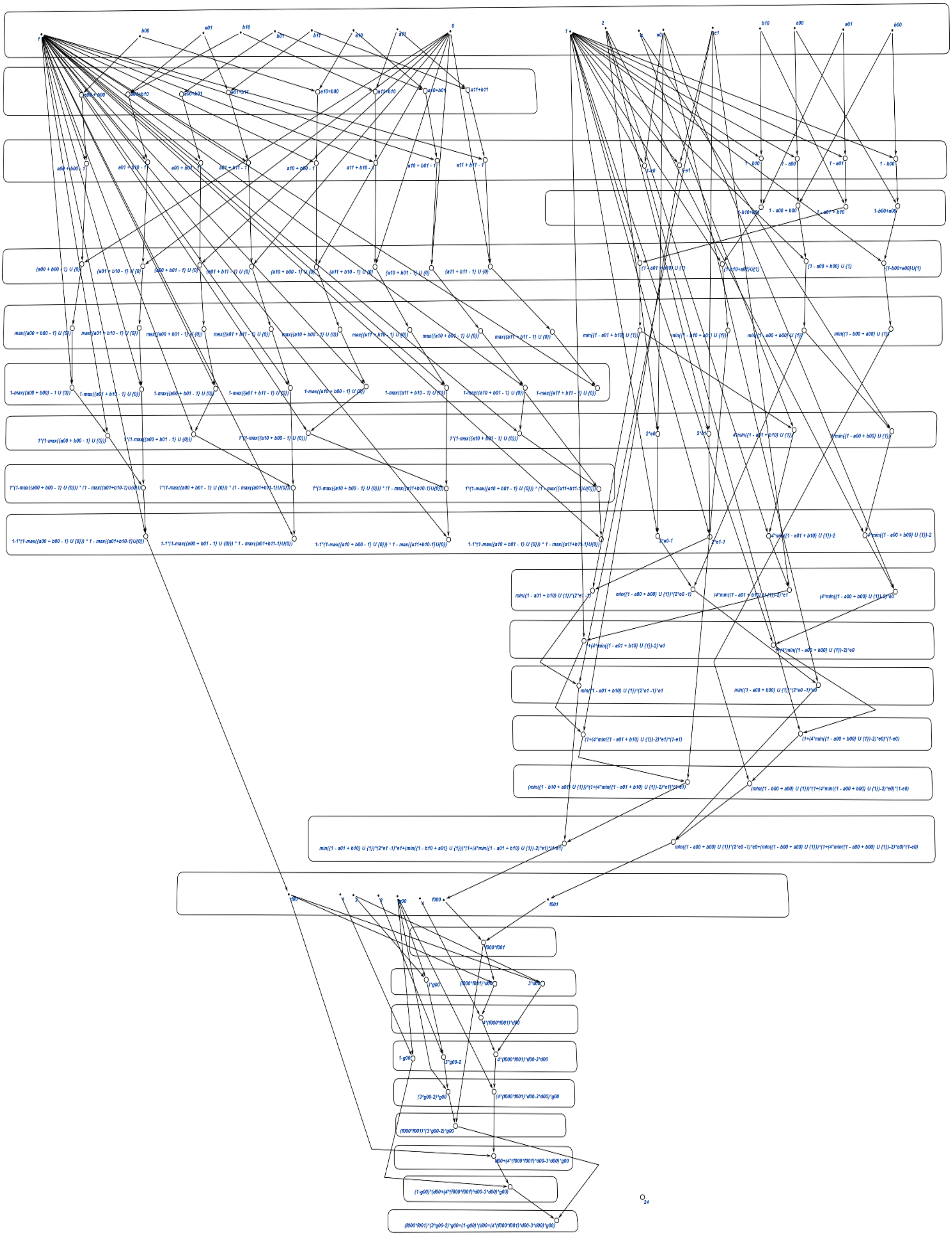
Зависимость коэффициента расхождения программы (D) от количества элементов (n)

При увеличении количества процессорных элементов, возрастает коэффициент расхождения программы

Зависимость коэффициента расхождения программы (D) от ранга задачи (r)

При увеличении ранга задачи, снижается значение коэффициента расхождения программы

4. Информационный граф:



Вывод:

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.