Федеральное государственное бюджетное образовательное учреждение высшего образования

«Сибирский государственный университет телекоммуникаций и информатики»

(СибГУТИ)

Кафедра ВС

РАСЧЕТНО-ГРАФИЧЕСКОЕ ЗАДАНИЕ

по дисциплине «Функциональное и логическое программирование»

Вариант 19

Выполнил: студент гр. ИП-612

Самусев М.Р.

Проверил: Ефимов А.В.

Новосибирск 2018 г.

**ОГЛАВЛЕНИЕ**

[1. ЗАДАНИЕ 3](#_Toc533270392)

[2. МОДЕЛЬ КОЛЛЕКТИВА ВЫЧИСЛИТЕЛЕЙ 3](#_Toc533270393)

[3. РАСЧЕТ ФУНКЦИЙ НАДЕЖНОСТИ И ГОТОВНОСТИ 5](#_Toc533270394)

[4. ПАРАЛЛЕЛЬНЫЙ АЛГОРИТМ УМНОЖЕНИЯ МАТРИЦ 9](#_Toc533270395)

[5. СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ 12](#_Toc533270396)

# 1. ЗАДАНИЕ

1. Выполнить анализ архитектурных принципов модели коллектива вычислителей. Привести пример суперВС, в которой модель используется на нескольких уровнях иерархической функциональной структуры.
2. Произвести численный расчет и построить графики для функций надежности *r*(*t*) и готовности *s*(*i*, *t*) ЭВМ, обладающей следующими техническими параметрами:

– средним временем безотказной работы 105 ч,

– интенсивностью восстановления 101/*ч*.

1. Построить блок-схему *p* -алгоритма умножения матриц:

*W*1:A; 1:B, *Y* 1:B; 1:C,

обеспечивающего распределение элементов результирующей матрицы по горизонтальным полосам в элементарных машинах ВС.

Отыскать максимум коэффициента накладных расходов при реализации

*p* -алгоритма на вычислительной системе, имеющей следующие параметры:

– разрядность *l* 32;

– полосу пропускания канала между машинами 100 Мегабод;

– время выполнения операции сложения *tc* нс;

– время выполнения операции умножения *ty* нс.

# 2. МОДЕЛЬ КОЛЛЕКТИВА ВЫЧИСЛИТЕЛЕЙ

Под коллективом вычислителей понимается совокупность вычислительных машин, программно-аппаратурным способом настраиваемая на решение общей задачи.

Модель представляется парой: S = <H, A> где H и A – описание конструкции (или просто: конструкция) и алгоритм работы коллектива вычислителей. Конструкция коллектива вычислителей описывается в виде: H = <C, G> где C={ci}– множество вычислителей сi (i =0..N-1), N– мощность множества C, G– описание макроструктуры коллектива вычислителей, т.е. структуры сети связей между вычислителями ci (или структура коллектива).

В основу коллектива вычислителей положены три основополагающих архитектурных принципа:

1. параллелизм (parallelism, concurrency) при обработке информации;
2. программируемость структуры (programmability, adoptability) — настраиваемости структуры сети связей между вычислителями, достигаемой программными средствами;
3. однородность конструкции (homogeneity), однородности вычислителей и структуры.

Модель коллектива вычислителей опирается на положения, противоположные принципам, лежащим в основе конструкции вычислителя. Принцип программируемости структуры является не менее фундаментальным в области архитектуры средств обработки информации, чем предложение Дж. фон Неймана по хранению программы работы ЭВМ в ее памяти и модификации программы с помощью самой же машины. Требования принципа программируемости структуры сводятся к тому, чтобы в коллективе вычислителей была заложена возможность хранения описания изначальной физической структуры, априорной автоматической (программной) настройки проблемно-ориентированных (виртуальных) конфигураций и их перенастройки в процессе функционирования с целью обеспечения адекватности структурам и параметрам решаемых задач и достижения эффективности при заданных условиях эксплуатации.

Конструктивная однородность модели коллектива вычислителей заключается в формировании его из совокупности одинаковых вычислителей, регулярно соединенных между собой. В отличие от одиночного вычислителя, коллектив вычислителей обладает теоретически неограниченной производительностью, обусловленной отсутствием ограничений на увеличение их числа. Кроме того, по сравнению с ЭВМ, построенное на модели коллектива вычислительное средство способно обладать заданной надежностью и живучестью, т.е. возможностью функционировать при отказах элементов, а также допускает простой способ наращивания производительности.

Уровень развития вычислительной математики и техники, а также технологии микроминиатюризации (микроэлектроники и наноэлектроники) уже сейчас позволяет в некоторых областях вместо принципа конструктивной однородности (однородности состава C и структуры G) использовать принцип квазиоднородности (или виртуальной однородности) конструкции H. Более того, можно ограничиться лишь требованием совместимости вычислителей в коллективе и использовать неоднородные структуры.

Вычислительное средство, базирующееся на модели коллектива вычислителей, называется вычислительной системой. На данный момент представляют вычислительные системы с массовым параллелизмом, такие системы называются СуперВС.

Одной из таких систем является CRAY T3D.

# 3. РАСЧЕТ ФУНКЦИЙ НАДЕЖНОСТИ И ГОТОВНОСТИ

Функция (или вероятность безотказной работы) относится к основным показателям надежности ЭВМ. Характеризует производительность ЭВМ на промежутке времени, то есть эта функция обеспечивает потенциально возможную производительность. Функцией надежности ЭВМ называется



где запись  означает вероятность того, что для всякого , принадлежащего промежутку времени  производительность  ЭВМ равна единице, т.е. равна потенциально возможной.

Функция  обладает следующими свойствами:

1.  Т.е. машина в момент начала функционирования находится в работоспособном состоянии.
2.  Событие, заключающееся в том, что ЭВМ работоспособна на конечном промежутке времени, является достоверным.
3.  для ;

Функцией ненадежности (или вероятностью отказа) ЭВМ называется



Функция  позволяет определить среднее время безотказной работы (средняя наработка до отказа). По определению, среднее время  безотказной работы ЭВМ и оценка  соответственно равны:



где – время безотказной работы -й машины, 

Интенсивностью отказов (лямбда-характеристикой) ЭВМ называется функция



Практически установлено, что зависимость интенсивности отказов от времени имеет место на периоде приработки ЭВМ. После приработки ЭВМ интенсивность отказов остается постоянной (до вхождения в предельное состояние или, по крайней мере, в течение промежутка времени, перекрывающего время морального старения). Следовательно, в нормальных условиях эксплуатации ЭВМ  а функция надежности и математическое ожидание времени *безотказной работы* соответственно равны:

.

– среднее число отказов, появляющихся в машине в единицу времени.

Подставляя известные нам данные получим следующую функцию для расчета надежности:

*r*(*t*)=exp(-1/\**t*)

*r*(*t*)=exp(-*t*/105);

Рассчитаем значения функции и построим график:

|  |  |
| --- | --- |
| *t, ч.* | *r(t)* |
| 0 | 1,000000 |
| 1 | 0,999990 |
| 5 | 0,999950 |
| 10 | 0,999900 |
| 100 | 0,999000 |
| 1000 | 0,990050 |
| 10000 | 0,904837 |
| 20000 | 0,818731 |
| 30000 | 0,740818 |
| 40000 | 0,670320 |
| 50000 | 0,606531 |
| 60000 | 0,548812 |
| 70000 | 0,496585 |
| 80000 | 0,449329 |
| 90000 | 0,406570 |
| 100000 | 0,367879 |
| 150000 | 0,223130 |
| 200000 | 0,135335 |
| 250000 | 0,082085 |
| 300000 | 0,049787 |
| 350000 | 0,030197 |
| 400000 | 0,018316 |
| 500000 | 0,006738 |
| 1000000 | 0,000045 |

Теперь рассчитаем значения функции готовности. Функция готовности ЭВМ

 

есть вероятность того, что (в условиях потока отказов и восстановлений) машина будет иметь в момент времени  производительность, равную единице, т.е. равную потенциально возможной.

Функция готовности ЭВМ обладает следующими свойствами:

1. 
2. 
3.   для 

Расчет будем производить по следующим формулам:

;

.

для начальных состояний ЭВМ , причем  соответствует состоянию отказа, а  – работоспособному состоянию машины, где *λ*=1/.

s (0, t)=10 / (10 + 1/105) – 10 / (10 + 1/105)\*exp((-t) \* (10 + 1/105))=

=0,999999 – 0,999999 \* exp((-t) \* 10,00001);

*s* (1, *t*)=10 / (10 + 1/105) + 0,000001 / (10 + 1/105)\*exp((-*t)* \* (10 + 1/105))=

=0,999999 + 0,00000099 \* exp((-t) \* 10,00001).

Рассчитаем значения функции и построим график:

|  |  |  |
| --- | --- | --- |
| *t*, ч. | *s*(0, *t*) | *s*(1, *t*) |
| 0 | 0,00000 | 0,99999999 |
| 0,001 | 0,00995 | 0,99999998 |
| 0,01 | 0,09516 | 0,99999990 |
| 0,05 | 0,39347 | 0,99999960 |
| 0,08 | 0,55067 | 0,99999944 |
| 0,1 | 0,63212 | 0,99999936 |
| 0,2 | 0,86466 | 0,99999913 |
| 0,3 | 0,95021 | 0,99999905 |

# 4. ПАРАЛЛЕЛЬНЫЙ АЛГОРИТМ УМНОЖЕНИЯ МАТРИЦ

***W*** [1:*A*; 1:*B*] ** ***Y*** [1:*B*; 1:*C*] = ***X*** [1:*A*; 1:*C*]

Количество столбцов *B* в матрице ***W*** равно количеству строк *B* матрице ***Y***.

Вычислитель 1

Вычислитель 2

…

Вычислитель *l*

…

Вычислитель *n*

Вычислитель 1

Вычислитель 2

…

Вычислитель *l*

…

Вычислитель *n*

Вычислитель 1

Вычислитель 2

…

Вычислитель *l*

…

Вычислитель *n*

 

***W***  ** ***Y*** = ***X***

Для построения p-алгоритма прежде всего требуется осуществить распределение исходного массива данных. Осуществим следующие распределения:

Матрицу ***W*** разобьем на n равных вертикальных полос, а матрицу ***Y*** на n равных горизонтальных полос.

**Начало**



**Конец**

***i*: = 1**

**Да**





**Нет**

**Прием**

**|| *w****1i****,…,w****hi****,..,w****Bi* ***||***

***l*** *= α*

***?***

**Нет**

**Да**

**Вычисление**

**Передача**

**|| *w****li****,…,w****hi****,…,w****Bi* ***||***

***i > α*]*A/n*[**

**?**

***i:= i + 1***

**Да**

**Нет**

]A / n[ (*l* - 1) < *i* ≤ ]A/ n[ *l* ,

***α*** – номер передающего вычислителя,

 – номера принимающих вычислителей.

Эффективность параллельного алгоритма умножения матриц большого размера можно характеризовать показателями:

******

Очевидно, что максимум накладных расходов будет при  (равенство  достигается при *n = A*). Таким образом, максимум коэффициента ε накладных расходов определяется формулой:

ε = *tn / (ty + tc)*

tn – время пересылки одного слова (элемента матрицы);

tу и tс – время выполнения операций умножения и сложения соответственно.

Подставляя свои значения *l* = 32, *ν* = 100 Мегабод, *tc* нс, *ty* нс, получаем:

tn = *l* / ν = 32 / 108 = 320 нс,

ε = 320 / (100 + 10) = 2,909

Ответ: максимум коэффициента накладных расходов при реализации *p* - алгоритма на данной вычислительной системе составляет 2,909.

# 5. СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

1. Хорошевский В.Г. Архитектура вычислительных систем. – М.: МГТУ им. Н.Э. Баумана, 2008. – 520 с.
2. Конспект лекций по курсу “Архитектура вычислительных систем”