



МИНОБРНАУКИ РОССИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Самарский государственный технический университет»
(ФГБОУ ВО «СамГТУ»)

Кафедра «Вычислительная техника»

ИНСТАЛЛЯЦИЯ И СПРЯЖЕНИЕ АППАРАТНО-ПРОГРАММНЫХ КОМПЛЕКСОВ

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ЛАБОРАТОРНЫМ РАБОТАМ
Для направления 09.03.01 «Информатика и вычислительная техника»
Профиль «Вычислительные машины, комплексы, системы и сети»

Самара 2017

Составитель Н.В.ЕФИМУШКИНА

УДК 681.324

Инсталляция и сопряжение аппаратно-программных комплексов: Метод. указ. к лаб. работам/ Самар. гос. техн. ун-т; Сост. *Н.В.Ефимушкина*. Самара, 2017. 62 с.

Дано руководство к четырем лабораторным работам по одному из курсов, изучаемых студентами направления 09.03.01 Информатика и вычислительная техника, профиля «Вычислительные машины, комплексы, системы и сети». Для каждой работы приведена краткая теория и необходимые расчетные формулы. Описан порядок выполнения работы, содержание отчета и дан список контрольных вопросов, позволяющих закрепить изучаемый материал.

Методические указания предназначены для студентов направления 09.03.01. Они могут быть использованы при исследовании и проектировании реальных вычислительных систем соответствующих классов.

Ил. 18. Табл. 12. Библиогр.: 3 назв.

Печатается по решению редакционно-издательского совета СамГТУ

Лабораторная работа №1

ОЦЕНКА ТРУДОЕМКОСТИ АЛГОРИТМОВ

Краткая теория

Цель работы. Определение параметров программ, необходимых для проектирования вычислительных систем, на которых они реализуются.

В лабораторной работе трудоемкость (вычислительная сложность) алгоритма определяется с точки зрения использования им ресурсов вычислительной системы. При этом оцениваются временные характеристики, которые считаются случайными величинами, описываемыми на уровне соответствующих моментов. В качестве ресурсов вычислительной системы рассматриваются ее основные устройства: процессор, устройства ввода с клавиатуры и вывода на печать, дисплеи, внешняя память на жестких и гибких дисках и т.д. Параметры алгоритма задаются с помощью укрупненной схемы алгоритма, отображающей порядок выполнения этапов счета и ввода-вывода (обращения к внешним устройствам). На схеме воспроизводится последовательность использования перечисленных ресурсов вычислительной системы. При этом применяются три основных оператора: функциональный (счета); перехода (ветвления) и ввода-вывода. На схеме алгоритма должны быть определены.

1. Среднее количество процессорных операций Θ , выполняемых за один прогон алгоритма.

2. Среднее количество обращений N_1, \dots, N_K к файлам F_1, \dots, F_K (устройствам ввода-вывода и внешним устройствам) за один прогон (среднее количество операций ввода-вывода).

3. Среднее количество информации g_1, \dots, g_k , передаваемой за одно обращение к файлам F_1, \dots, F_K соответственно (среднее время выполнения операций ввода-вывода).

4. Вероятности p_{ij} перехода от оператора i к оператору j .

Значение Θ характеризует трудоемкость обработки информации (счета), а N_k , $k=1, \dots, K$ и g_k , $k=1, \dots, K$ — трудоемкость процесса ввода-вывода информации.

Последовательность операторов схемы алгоритма воспроизводит режим обработки задач, используемый данной ВС и реализуемый средствами ее операционной системы. Так, в системах коллективного пользования (локальной сети) на основе ПЭВМ типа IBM, работающих под управлением сетевой ОС Novell NetWare, Windows NT и др., принят следующий порядок выполнения программ (при вводе и отладке нового приложения на языке высокого уровня).

1. Ввод имени и пароля с устройства ввода (дисплей).

2. Контроль правильности задания имени и пароля, при наличии ошибки — завершение работы или повторная попытка пока не надоест (основное устройство - процессор сервера).

3. Если при входе в систему не допущена ошибка, то происходит вызов редактора языка программирования и ввод текста программы надоест (Основное устройство - жесткий диск).

4. Сохранение набранного текста в рабочем каталоге на винчестере.

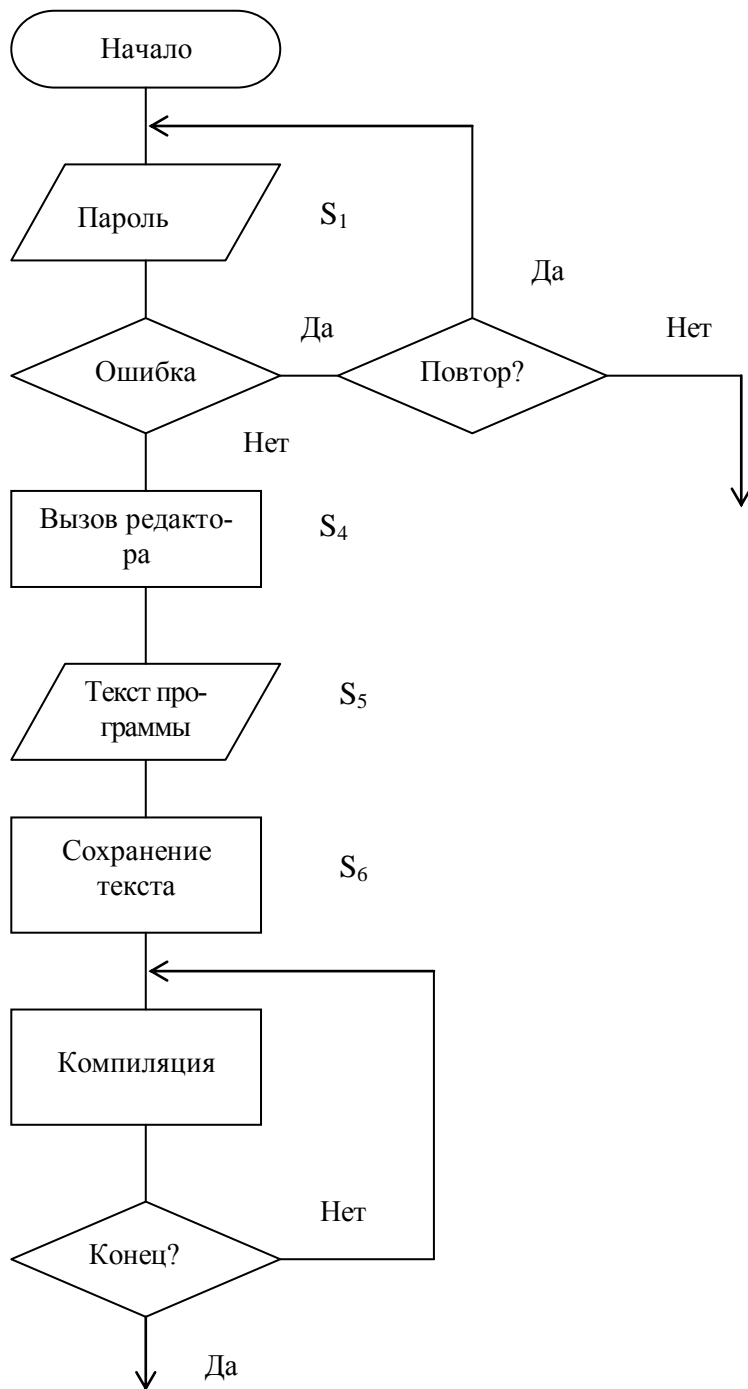


Рис. 1.1. Фрагмент схемы алгоритма работы с новой программой в системе коллективного пользования

фрагмент графа Марковской модели, соответствующей схеме алгоритма рис.1.1, приведен на рис. 1.2.

Марковская модель считается заданной, если заданы:

5. Выполнение прохода компиляции (при n -проходном компиляторе выполняется n циклов процессорной обработки с переписью промежуточных результатов на винчестер).

И так далее.

Фрагмент соответствующей укрупненной схемы алгоритма представлен на рис. 1.1.

Для оценки трудоемкости алгоритма в лабораторной работе используются Марковские модели вычислительных процессов. Модель строится по укрупненной схеме алгоритма путем замены каждой вершины схемы, кроме начальной и конечной, состоянием Марковской

- 1) ее состояния S_1, \dots, S_n ;
- 2) средние времена пребывания в каждом состоянии ν_1, \dots, ν_n ;
- 3) матрица $P = [p_{ij}]$ вероятностей переходов, которая определяет соответствующие вероятности для любых пар состояний $S_i - S_j$.

В матрице, кроме основных, добавляется еще одно, фиктивное состояние S_0 , которое имитирует внешнюю среду (начало и окончание выполнения алгоритма).

При построении модели вычислительного процесса состояниям S_1, \dots, S_n соответствуют, как отмечалось выше, блоки схемы алгоритма. Времена ν_1, \dots, ν_n пребывания в этих состояниях задаются с помощью трудоемкостей $\Theta_1, \dots, \Theta_n$ (количества выполняемых операций) и быстродействия устройств, B_1, \dots, B_m , на которых они выполняются. Основные исходные данные для соответствующих вариантов заданий, выполняемых в лабораторной работе, приведены в прил. 1.

Вероятности p_{ij} определяются так. Если в модели нет перехода из состояния S_i в состояние S_j , то $p_{ij} = 0$. Для линейных участков вероятности равны 1, а для ошибок выбираются по заданию преподавателя из диапазона: 0 — 0.5. Ветвления на циклических участках алгоритмов с

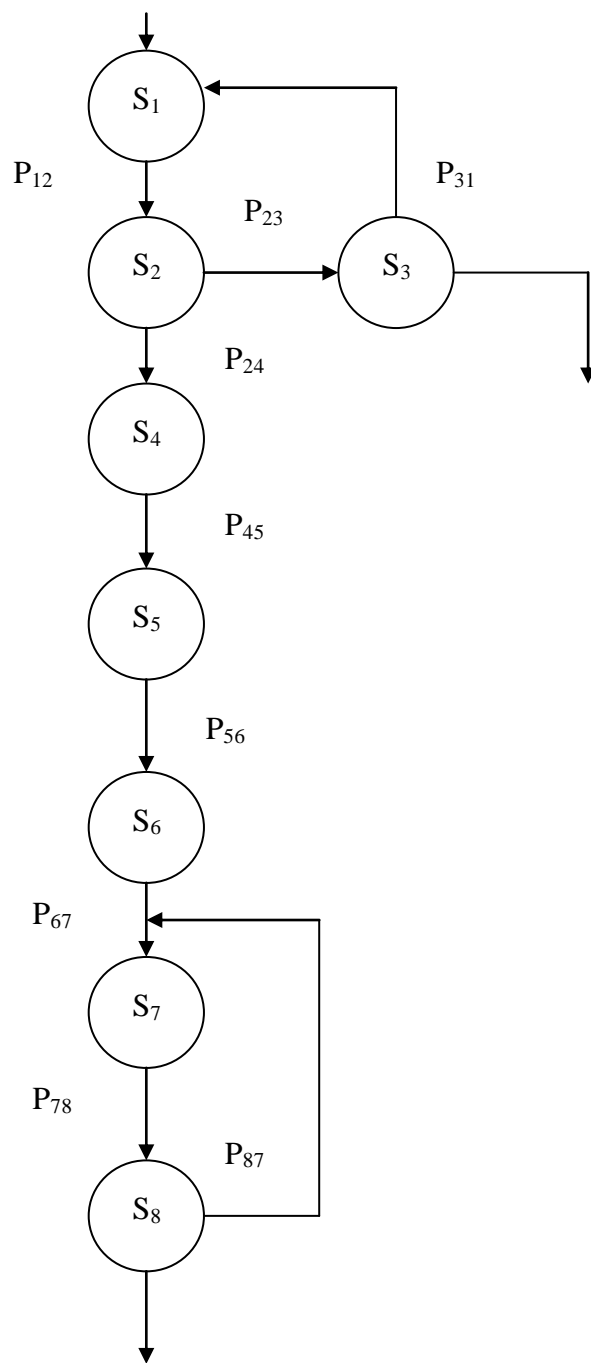


Рис. 1.2. Граф Марковской модели

известным количество повторений цикла m характеризуются вероятностями выхода из цикла

$$p_o = 1/m \quad (1.1)$$

и продолжения цикла

$$p_u = (m-1)/m. \quad (1.2)$$

Описание лабораторной установки

Лабораторная работа выполняется на ПЭВМ типа IBM, работающей в среде Windows 98/2000/NT и др. Программа составлена с использованием среды Delphi и позволяет сделать следующее:

- 1) ввести или скорректировать параметры устройств, на которых реализуется вычислительный процесс, порождаемый задачей, и сохранить эти параметры в файле;
- 2) ввести параметры Марковской модели;
- 3) оценить трудоемкость заданного алгоритма с помощью модели.

Исходные данные и результаты расчетов могут быть сохранены в файле текущего каталога на винчестере и затем распечатаны стандартными средствами.

Порядок выполнения лабораторной работы

Подготовка к работе

Подготовка предполагает выполнение следующих этапов.

1. Знакомство со всеми разделами руководства.

2. Получение у преподавателя задания на оценку трудоемкости алгоритма для определенного режима работы вычислительной системы (см. прил. 1).

3. Составление схемы алгоритма и определение вероятностей перехода p_{ij} по формулам (1.1) и (1.2) или заданию преподавателя.

В лабораторной работе по указанию преподавателя должны быть исследованы одна — две схемы алгоритма.

Последовательность выполнения лабораторной работы

Приложение имеет имя Markmod.exe и запускается стандартным образом. Ввод исходных данных возможен из файла (если пользователь работает с программой не первый раз) или с клавиатуры в режиме диалога. В первом случае используется кнопка «Загрузить», расположенная в нижней части окна приложения. Ввод исходных данных с клавиатуры или их коррекция осуществляется в соответствующих таблицах окна. Причем **задание параметров устройств** производится после нажатия **правой кнопки мыши**. Исходные данные и результаты расчетов могут быть сохранены в файлах, имена которых задаются пользователем. Указанная операция выполняется после нажатия кнопки «Сохранить», расположенной в нижней части окна.

При выполнении лабораторной работы студенты по заданию преподавателя должны получить значения трудоемкости алгоритмов для различных режимов работы в системе коллективного пользования. При этом необходимо исследовать влияние величин вероятностей ошибок, количества повторений циклов и количеств операций, выполняемых в отдельных вершинах, на трудоемкость алгоритма в целом и времена выполнения его ветвей.

Рекомендуется следующая последовательность работы с программой.

1. Задание параметров устройств, выполняющих обслуживание задачи (наименований и быстродействия). Оно осуществляется после нажатия правой кнопки мыши.

2. Ввод информации о параметрах вершин схемы алгоритма (состояниях Марковской модели).

3. Занесение значений вероятностей в матрицу P (в окне она названа таблицей).

4. Расчет характеристик алгоритма (начинается после нажатия на кнопку «Пуск») и сохранение их в файле.

5. Повторение п.п. 2 — 4 для различных значений трудоемкостей вершин схемы алгоритма и вероятностей основных переходов (по заданию преподавателя и диапазонам значений (см. прил. 1)).

Содержание отчета о выполненной работе

Отчет о выполненной работе должен содержать.

1. Название и цель работы.
2. Схему предложенного варианта алгоритма.
3. Граф Марковской модели.
4. Перечень используемых устройств и их характеристики.

5. Матрицу вероятностей переходов для первого набора исходных данных, а также, по-возможности, распечатки исходных данных и результатов. Если получить последние по какой-либо причине не удалось, то они должны быть сохранены в файле и продемонстрированы при защите отчета.

Контрольные вопросы

1. Чем характеризуется трудоемкость алгоритма?
2. От чего зависит структура схемы алгоритма, чем определяется уровень ее детализации?
3. Как строится модель вычислительного процесса, порождаемого алгоритмом?
4. Какие блоки схемы исследованного вами алгоритма вносят наибольший вклад в значение трудоемкости и почему?
5. Как влияет вероятность ошибок в различных частях схемы на время выполнения алгоритма?
6. Как влияет количество циклов в различных частях схемы на время выполнения алгоритма?
7. Можно ли оценить трудоемкость по обычным схемам алгоритмов, используемым в программировании или микропрограммировании?
8. Как использовать результаты лабораторной работы при исследовании вычислительных систем?
9. Каковы основные свойства Марковских моделей, их параметры и характеристики?
10. Как влияют характеристики устройств на трудоемкость алгоритма?

Лабораторная работа №2

АНАЛИЗ ХАРАКТЕРИСТИК ФУНКЦИОНИРОВАНИЯ ЦУС

Краткая теория

Цель работы. Анализ зависимостей характеристик функционирования ЦУС от быстродействия процессора, параметров задач, решаемых системой, и дисциплины обслуживания задач.

Особенности работы и характеристики цифровых управляющих систем

Цифровые управляющие системы (ЦУС) — это системы, работающие в темпе, задаваемом объектом управления, т.е. в так называемом реальном масштабе времени (РМВ). Примером таких систем могут служить системы управления энергетическими объектами, телефонным и телеграфным оборудованием, автоматизации измерений, АСУ и т. д. ЦУС соединяются с объектом множеством каналов ввода-вывода, по которым поступают сигналы, характеризующие объект, и передаются управляющие воздействия. Типовая схема ЦУС на базе однопроцессорной ЭВМ приведена на рис. 2.1, а схема ее функционирования — на рис. 2.2.

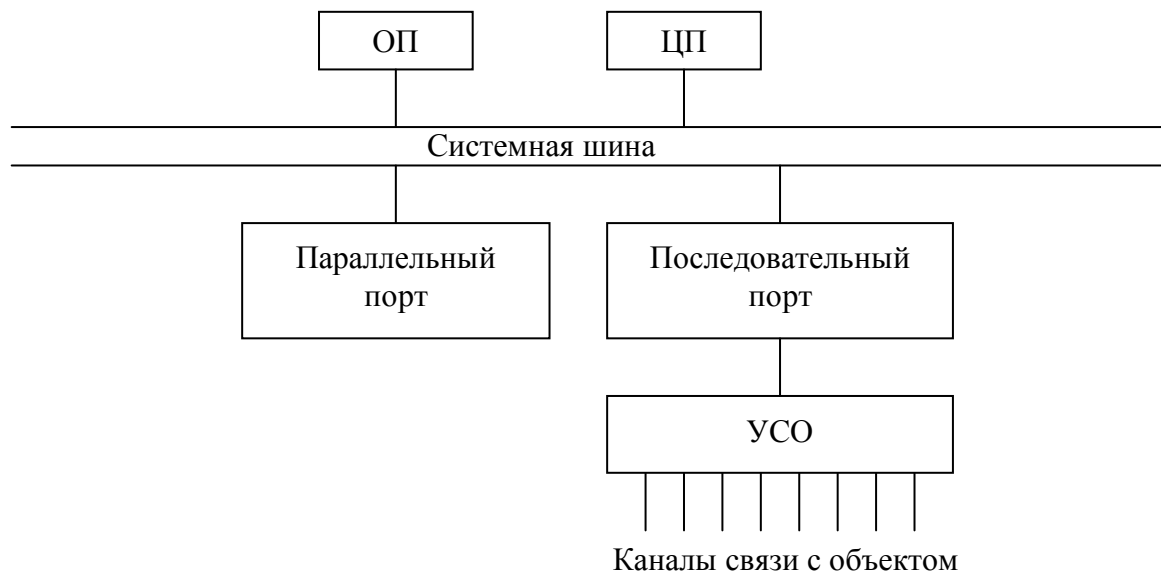


Рис. 2.1. Типовая структура ЦУС: ОП – оперативная память, ЦП – центральный процессор, УСО – устройство связи с объектом

Основными особенностями работы описываемых систем являются:

- функционирование в реальном масштабе времени, т. е. наличие предельных ограничений на время решения задач (время ответа);

- ограниченное количество программ Π_1, \dots, Π_m , хранящихся в оперативной памяти ЭВМ;
- необходимость обеспечения высокой программной устойчивости и надежности.

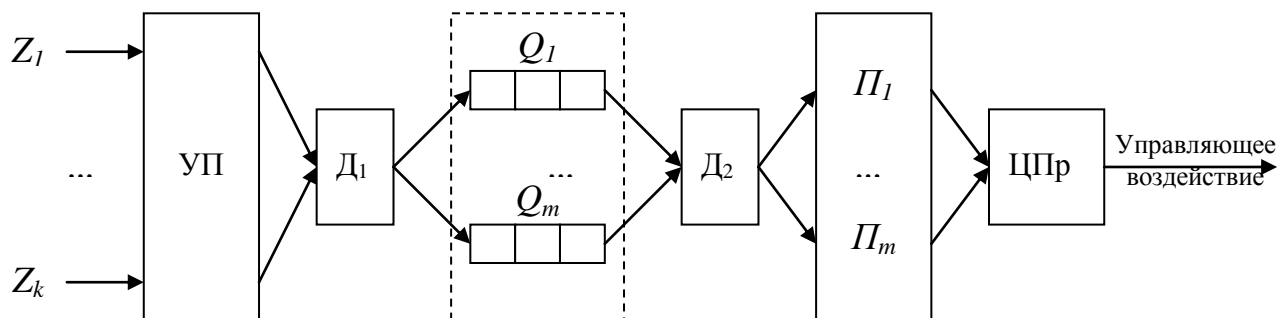


Рис. 2.2. Схема функционирования цифровой управляющей системы:

УП – устройство прерывания; D_1, D_2 – программы-диспетчеры;

Z_1, \dots, Z_k – заявки на решение задач; Q_1, \dots, Q_m – очереди заявок;

Π_1, \dots, Π_m – программы; ЦПр – центральный процессор

ЦУС работает так. Периодически или в случайные моменты времени по каналам ввода-вывода поступают заявки Z_1, \dots, Z_k на устройство прерывания, которое инициирует программу диспетчер D_1 . Эта программа распределяет заявки по очередям Q_1, \dots, Q_m . Затем программа D_2 в соответствии с реализуемой дисциплиной обслуживания выбирает одну заявку из очереди и посылает ее на обслуживание в центральный процессор. Каждой заявке соответствует своя программа обработки Π_1, \dots, Π_m . Они вырабатывают соответствующие управляющие воздействия на объект. Заявки Z_1, \dots, Z_k образуют поток, который характеризуется следующими параметрами:

1) количеством классов задач M (при $M=1$ поток считается одномерным, при $M > 1$ – многомерным);

2) интенсивностью λ - средним количеством заявок, поступающих в систему в единицу времени

$$\lambda = n/t, c^{-1};$$

3) интенсивностью обслуживания $\mu = 1/v, c^{-1};$

где v - среднее время обслуживания (решения задач в системе) без учета ожидания в очереди;

4) коэффициентом вариации среднего времени обслуживания

$$v^2 = D[t_{обсл}] / v^2,$$

где $D[t_{обсл}]$ – дисперсия времени обслуживания $t_{обсл}$.

При одномерном потоке заявок работа цифровой управляющей системы описывается однородной одноканальной СМО вида рис. 2.3 и имеет следующие характеристики:

1) среднее время пребывания заявок в системе (время ответа)

$$u = v + w,$$

где v - среднее время обслуживания (решения задач в ЦУС); w – среднее время ожидания в очереди. Качество функционирования ЦУС характеризуется, прежде всего, временем ожидания w , так как среднее время обслуживания v определяется алгоритмом задачи и не зависит от организации работы системы;

2) загрузка системы или среднее количество заявок, находящихся на обслуживании:

$$\rho = \lambda v.$$

В реальных вычислительных системах $0 \leq R \leq 1$. В этом случае в них имеет место стационарный режим. При $R > 1$ система перегружена, и очередь заявок растет до бесконечности;

3) коэффициент простоя системы $\eta = 1 - \rho;$

4) средняя длина очереди заявок $l = \lambda w;$

5) среднее количество заявок, находящихся в системе: $n =$

$$\lambda u = \rho + l.$$

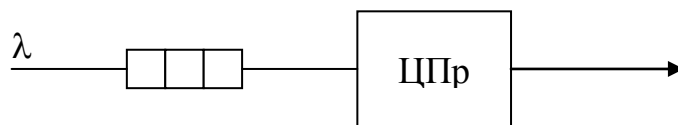


Рис. 2.3. Модель ЦУС при одномерном потоке заявок

При многомерном потоке заявок с интенсивностями поступления $\lambda_1, \dots, \lambda_M$, суммарной интенсивностью

$$\Lambda = \sum_{m=1}^M \lambda_m$$

и интенсивностями обслуживания μ_1, \dots, μ_M моделью ЦУС является неоднородная СМО вида рис. 2.4. Характеристики системы для отдельных потоков определяются по выше приведенным формулам (для однородного потока). Общие показатели системы вычисляются как сумма характеристик отдельных потоков:

1) загрузка

$$R = \sum_{m=1}^M R_m;$$

2) коэффициент простоя

$$h = 1 - R;$$

3) средняя длина очереди

$$L = \Lambda * W = \sum_{m=1}^M \lambda_m w_m;$$

4) среднее количество заявок в системе

$$N = \Lambda * U = \sum_{m=1}^M \lambda_m u_m;$$

5) временные характеристики суммируются с учетом вероятностей $p_m = \lambda_m / \Lambda$ появления заявок m -того класса в общем потоке:

а) среднее время ответа

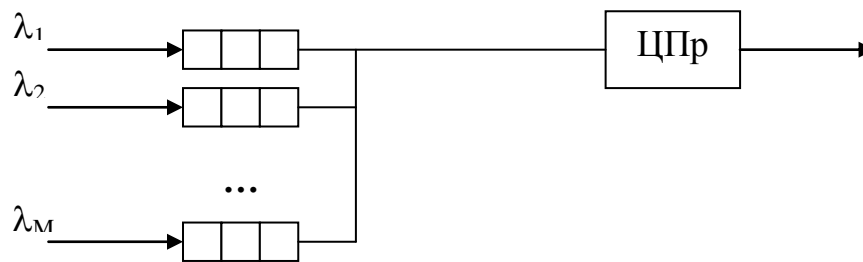
$$U = \sum_{m=1}^M p_m u_m;$$

б) среднее время ожидания

$$W = \sum_{m=1}^M p_m w_m;$$

в) среднее время обслуживания

$$V = \sum_{m=1}^M p_m v_m.$$



Р и с . 2.4. Модель ЦУС при многомерном потоке заявок

Очевидно, что время ожидания заявок зависит от порядка их выбора из очереди (дисциплины обслуживания). При этом оказывается, что уменьшение времени ожидания заявок одного класса достигается за счет увеличения этого времени для других классов, т. е., имеет место закон сохранения времени ожидания

$$W * R = const .$$

Дисциплины обслуживания заявок (задач) делятся на три вида:

- 1) беспriorитетные (FIFO, LIFO, RANDOM);
- 2) с относительными и абсолютными приоритетами;
- 3) смешанные, являющиеся комбинацией первых двух.

Приоритет – это преимущественное право на обслуживание. Его принято обозначать номерами $1, 2, \dots, M$ (от высшего к низшему). При абсолютных приоритетах заявка высшего приоритета может прервать обслуживание заявки низшего, а при относительных происходит обслуживание без прерывания. Для удобства описания смешанных дисциплин используются графические схемы. На этих схемах потоки заявок с одинаковыми приоритетами изображаются в виде прямоугольника, в котором записаны номера потоков (рис. 2.5, а). Отдельные прямоугольники соединяются между собой стрелками, направленными от более приоритетных потоков к менее приоритетным. Для отображения

относительных приоритетов используется пунктирная стрелка (рис. 2.5, б), а для абсолютных — сплошная (рис. 2.5, в). Смешанная дисциплина обслуживания может быть представлена схемой (рис. 2.5, г), на которой потоки 1, 2, 4 и 6 имеют абсолютные приоритеты по отношению к потокам 3 и 5. Внутри этих множеств между потоками 1, 4 и 2, 6, а также потоками 3 и 5 установлены относительные приоритеты. Заявки потоков 1 и 4 (2 и 6) не имеют приоритетов по отношению друг к другу.

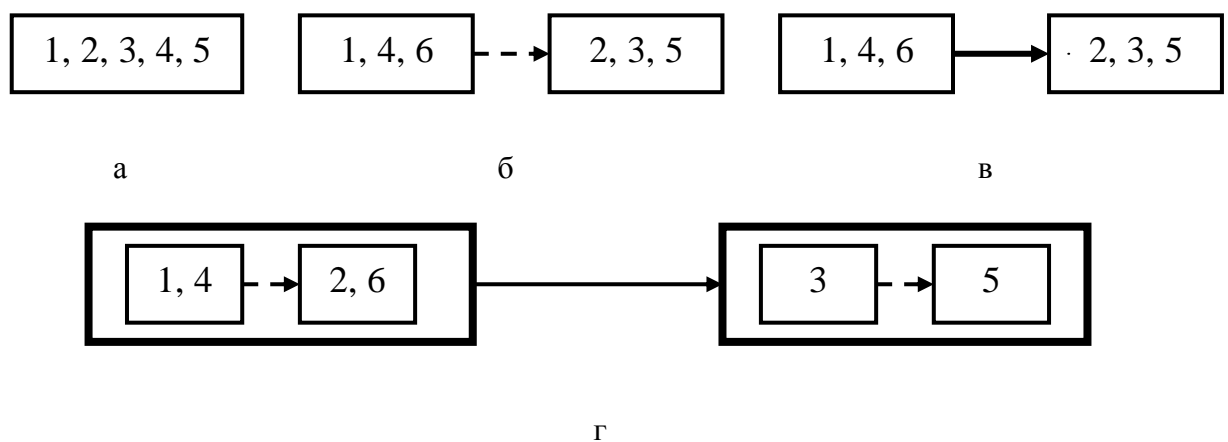


Рис. 2.5. Графические схемы распределения приоритетов: а - одинаковые приоритеты; б - относительные; в - абсолютные; г - смешанные приоритеты.

Для формального описания дисциплин обслуживания заявок и определения характеристик ЦУС с помощью ЭВМ используются матрицы приоритетов $Q=[q_{i,j}]$ размерностью $M*M$. Элементы матрицы: $q_{i,j} \in \{0, 1, 2\}$, где:

1) $q_{i,j} = 0$, если заявки потока i не имеют приоритета по отношению к заявкам потока j ;

2) $q_{i,j} = 1$, если заявки потока i имеют более высокий относительный приоритет по отношению к заявкам потока j ;

3) $q_{i,j} = 2$, если между заявками потоков i и j имеется абсолютный приоритет.

В матрице Q элементы главной диагонали равны 0, так как между заявками одного и того же потока не могут быть установлены никакие приоритеты. Очевидно, что если $q_{i,j}$ равно 1 или 2, то $q_{j,i}$ равно 0, потому что в этом случае заявки потока j не имеют приоритета по отношению к заявкам потока i . Для дисциплины FIFO матрица Q является нулевой, т. е. имеет все нулевые элементы. В случае относительных или абсолютных приоритетов, значения которых совпадают с номерами потоков, элементы $q_{i,j}$ равны 1 или 2 соответственно для всех $i > j$, т. е. матрица Q является диагональной вида

$$Q = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 1 & 1 & 1 \\ 2 & 0 & 0 & 1 & 1 \\ 3 & 0 & 0 & 0 & 1 \\ 4 & 0 & 0 & 0 & 0 \end{array} \quad Q = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 2 & 2 & 2 \\ 2 & 0 & 0 & 2 & 2 \\ 3 & 0 & 0 & 0 & 2 \\ 4 & 0 & 0 & 0 & 0 \end{array}$$

Для дисциплины обслуживания со смешанными приоритетами заполнение матрицы удобно выполнять с помощью графических схем. Рассмотрим последовательность заполнения матрицы приоритетов в этом случае на примере схемы рис. 2.5, г. Перед заполнением матрица Q имеет вид, представленный на рис. 2.6,а, т. е. содержит нули на главной диагонали. Заполнение матрицы начинается с заявок наивысшего приоритета — потоков 1 и 4.

Критерии эффективности и задача синтеза ЦУС

Основными критериями эффективности цифровых управляющих систем являются функции штрафа вида

$$C_w = \sum_{m=1}^M \alpha_m \lambda_m w_m,$$

где α_m — штраф за задержку задачи m -того класса;

λ_m — интенсивность поступления задач этого класса;

w_m – среднее время ожидания.

При одинаковых штрафах ($\alpha_m = const$) величина C_w равна длине очереди, т.е.

$$C_w = \sum_{m=1}^M \lambda_m w_m = L.$$

	1	2	3	4	5	6	
1	0						
2		0					
3			0				
4				0			
5					0		
6						0	

а

	1	2	3	4	5	6	
1	0	1	2	0	2	1	
2	0	0		0			
3	0		0	0			
4	0	1	2	0	2	1	
5	0			0	0		
6	0			0		0	

в

	1	2	3	4	5	6	
1	0	1	2	0	2	1	
2	0	0	2	0	2	0	
3	0	0	0	0		0	
4	0	1	2	0	2	1	
5	0	0	0	0		0	
6	0	0	2	0	2	0	

д

	1	2	3	4	5	6	
1	0	1	2	0	2	1	
2	0	0					
3	0		0				
4	0			0			
5	0				0		
6	0					0	

б

	1	2	3	4	5	6	
1	0	1	2	0	2	1	
2	0	0	2	0	2	0	
3	0	0	0	0			
4	0	1	2	0	2	1	
5	0	0		0	0		
6	0	0		0		0	

г

	1	2	3	4	5	6	
1	0	1	2	0	2	1	
2	0	0	2	0	2	0	
3	0	0	0	0	1	0	
4	0	1	2	0	2	1	
5	0	0	0	0	0	0	
6	0	0	2	0	2	0	

е

Рис. 2.6. Пример заполнения матрицы приоритетов

Другой более мощный критерий учитывает потери от простоя процессора

$$C_B = \alpha_0 \eta(B) + \sum_{m=1}^M \alpha_m \lambda_m w_m ,$$

где $\eta(B)$ – коэффициент простоя процессора, а α_0 – штраф за простой.

При проектировании ЦУС функция C_w используется для выбора дисциплины обслуживания задач, а C_B – для выбора процессора.

Для большинства цифровых управляющих систем задаются ограничения на среднее время пребывания или ожидания заявок (задач), которые представляются в виде

$$U_m \leq U_m^* \text{ или } W_m \leq W_m^*,$$

где U_m^* – допустимое время пребывания;

W_m^* – допустимое время ожидания.

Выполнение этих ограничений может привести к тому, что критерии C_w и C_B не будут минимальными. Очевидно, что существует дисциплина обслуживания с быстродействием процессора B_{min} , при котором ограничения на время ожидания или пребывания выполняются для всех типов заявок. При этом критерием эффективности цифровой управляющей системы будет коэффициент дисциплины обслуживания

$$\delta = (B - B_{min}) / B_{min}$$

где B – быстродействие процессора для выбранной дисциплины. Чем меньше δ , тем лучше дисциплина.

При синтезе ЦУС ищутся такие структура и дисциплина обслуживания, которые обеспечивают оптимум приведенных критериев. Задача синтеза ЦУС подразделяется на три подзадачи:

- 1) определение нижней оценки быстродействия процессора;
- 2) выбор дисциплины обслуживания;
- 3) определение оптимального для выбранной дисциплины быстродействия процессора.

Первая подзадача решается с использованием условия существования в системе стационарного режима

$$R = \sum_{m=1}^M \lambda_m V_m < 1,$$

где $V_m = \Theta_m / B$,

Θ_m - среднее количество команд у задач класса m .

Минимальное быстродействие определяется из неравенства вида

$$B_{\min} > \sum_{m=1}^M \Theta_m \lambda_m.$$

Для ЦУС с ограничениями на времена ожидания или пребывания B_{\min} должно быть выше.

Выбор дисциплины обслуживания связан с решением трех задач.

1. Выбор режима обслуживания (*одиночный* – диспетчер обслуживает каждую заявку, *групповой* – обслуживание всей очереди от начала до конца);

2. Выбор класса приоритетов (без приоритета, относительные, абсолютные или смешанные приоритеты);

3. Распределение приоритетов между классами заявок.

При решении двух последних задач считают, что одна дисциплина лучше другой, если значение C_w для нее меньше. Значение этого критерия может быть записано в явном виде при экспоненциальном обслуживании заявок. Если приоритеты назначены так, что

$$\frac{\alpha_1}{v_1} \geq \frac{\alpha_2}{v_2} \geq \dots \geq \frac{\alpha_M}{v_M},$$

то C_w будет минимальным.

При одинаковых для всех заявок значениях штрафов α_m приоритеты должны назначаться в порядке возрастания среднего времени обслуживания: наивысший приоритет – коротким задачам, низший – длинным. Если заданы ограничения U^* (W^*) на время пребывания (ожидания) задач в системе, то приоритеты следует назначать в порядке возрастания этих значений. При этом минимизируется среднее запаздывание относительно указанного ограничения. Для смешанных

дисциплин обслуживания приоритеты назначаются направленным перебором. При этом оптимальной считается дисциплина, у которой относительные отклонения

$$\delta_m = (U_m^* - U_m) / U_m^*$$

примерно одинаковы для всех потоков заявок. Если величина δ_m для некоторого потока сильно отличается от других, то для этого типа заявок следует изменить значение, а иногда и класс приоритета (относительные, абсолютные). Определение оптимального быстродействия процессора выполняется с использованием критерия C_B для выбранной дисциплины обслуживания. Оптимальным считается такое быстродействие, которое обеспечивает заданное качество обслуживания, т. е. выполнение ограничений на времена с минимумом затрат (коэффициент загрузки системы должен быть максимален, а коэффициент простоя – минимален). Для ЦУС без ограничений на время ответа оптимальное быстродействие равно минимальному.

Описание лабораторной установки

Лабораторная работа выполняется на ПЭВМ типа IBM, работающей в среде Windows 98/2000/NT и др. Программа составлена с использованием среды Delphi и позволяет сделать следующее:

- 1) выбрать номер варианта и ввести параметры задач (заявок), хранящиеся в файле исходных данных;
- 2) задать дисциплину обслуживания задач (заявок) в виде матрицы приоритетов;
- 3) определить минимально необходимое для указанного набора задач быстродействие центрального процессора и оптимальное быстродействие для заданной дисциплины обслуживания;

4) определить характеристики ЦУС для каждого потока заявок и общие характеристики системы при заданной дисциплине обслуживания и оптимальном для этой дисциплины быстродействии процессора.

Программа позволяет оценить характеристики ЦУС при количестве потоков заявок не более 10. Результаты расчетов могут быть сохранены в файле на винчестере и затем распечатаны стандартными средствами.

Порядок выполнения лабораторной работы

Подготовка к работе

Подготовка предполагает выполнение следующих этапов.

1. Знакомство со всеми разделами руководства.
2. Получение у преподавателя задания на исследование ЦУС с конкретными потоками заявок. Варианты заданий приведены в табл. П.2.1, а характеристики потоков задач, перечисленных в этих вариантах, - в табл. П.2.2.
3. Выбор вариантов дисциплин обслуживания заявок и составление матрицы приоритетов для этих вариантов.

В лабораторной работе необходимо исследовать следующие дисциплины обслуживания:

- 1) в порядке поступления;
- 2) с относительными приоритетами;
- 3) с абсолютными приоритетами;
- 4) со смешанными приоритетами (2-3 варианта). Приоритеты потоков заявок следует назначать в соответствии с рекомендациями, приведенными в теоретическом разделе.

Последовательность выполнения лабораторной работы

Программа имеет имя Dcs.exe, находится в папке Лаб2_ЦУС и запускается стандартным образом. После запуска открываются два окна, в первом из которых приведены характеристики задач, обрабатываемых системой, а во втором – варианты заданий. Каждый вариант задается вычислительной нагрузкой, создаваемой шестью классами задач. После нажатия на кнопку «Пуск» верхнего меню необходимо ввести свою фамилию и номер варианта. Программа сама выбирает из первой таблицы характеристики задач по этому номеру. Затем пользователю предлагается задать дисциплину обслуживания заявок, т.е. ввести матрицу приоритетов. После нажатия на клавишу «ОК» программа выполняет необходимые расчеты и выводит окно для сохранения результатов. Они записываются в файл, имя которого задает пользователь, и выводятся на экран для анализа. Результаты из файла могут быть распечатаны традиционным способом.

Для продолжения работы с программой вновь нажимается кнопка «Пуск» и либо подтверждается имя пользователя и вариант задания, либо вводятся новые. В первом случае имеется возможность задать новую матрицу приоритетов.

При выполнении лабораторной работы студенты по заданию преподавателя должны получить характеристики ЦУС для следующих дисциплин обслуживания.

1. FIFO.
2. Относительные приоритеты, которые назначаются по:
 - а) порядку расположения потоков в варианте задания;
 - б) значениям трудоемкости задач;
 - в) значениям допустимого времени пребывания задач в системе.
3. Абсолютные приоритеты, которые назначаются по:

- а) порядку расположения потоков в варианте задания;
- б) значениям трудоемкости задач;
- в) значениям допустимого времени пребывания задач в системе.

4. Смешанные дисциплины обслуживания, у которых 2 потока имеют абсолютные приоритеты, 2 — относительные приоритеты и 2 потока приоритетов не имеют.

Затем, начиная с последней из рассмотренных дисциплин обслуживания, выполнить выравнивание по всем потокам относительных отклонений допустимого времени пребывания.

Содержание отчета о выполненной работе

Отчет должен содержать следующее:

- название и цель работы, исходные данные;
- распечатки результатов расчетов, выполненных на ЭВМ;
- графики зависимостей среднего времени ожидания и пребывания, коэффициентов загрузки и длин очередей для заявок отдельных потоков (номера потоков задаются преподавателем) от исследованных дисциплин обслуживания с указанием оптимальной дисциплины;
- выводы о результатах исследования ЦУС и наилучших и наихудших режимах функционирования.

Контрольные вопросы

1. Что такое ЦУС и каковы особенности их функционирования?
2. Что такое заявки и чем она характеризуется?
3. Какие модели применяются для оценки характеристик ЦУС?
4. Каковы основные характеристики качества функционирования ЦУС при одномерном потоке заявок?

5. Каковы основные характеристики качества функционирования ЦУС при многомерном потоке заявок?
6. Какие дисциплины обслуживания заявок исследуются в лабораторной работе?
7. Как можно представить распределение приоритетов заявок?
8. Опишите способ заполнения матрицы приоритетов.
9. Как назначаются приоритеты при различных дисциплинах обслуживания?
10. Какие рекомендации можно дать по выбору процессора и организации функционирования ЦУС для предложенного вам набора потоков задач?
11. Как начать и закончить сеанс работы с программой исследования ЦУС на ЭВМ?

Лабораторная работа №3

Синтез систем оперативной обработки

Краткая теория

Цель работы. Определение оптимальной структуры системы оперативной обработки, обеспечивающей обработку заданной рабочей нагрузки с минимальным средним временем ответа при ограничении на стоимость системы или имеющую минимальную стоимость при ограничении на время ответа.

Система оперативной обработки

Система оперативной обработки (СОО) — это вычислительная система, функционирующая в диалоговом режиме или режиме разделения времени. Примером таких систем могут служить различные информационно-поисковые системы, вычислительные системы и центры коллективного пользования, сети и т. д. Типовая структура СОО, соответствующая архитектуре систем фирмы IBM, приведена на рис. 3.1.

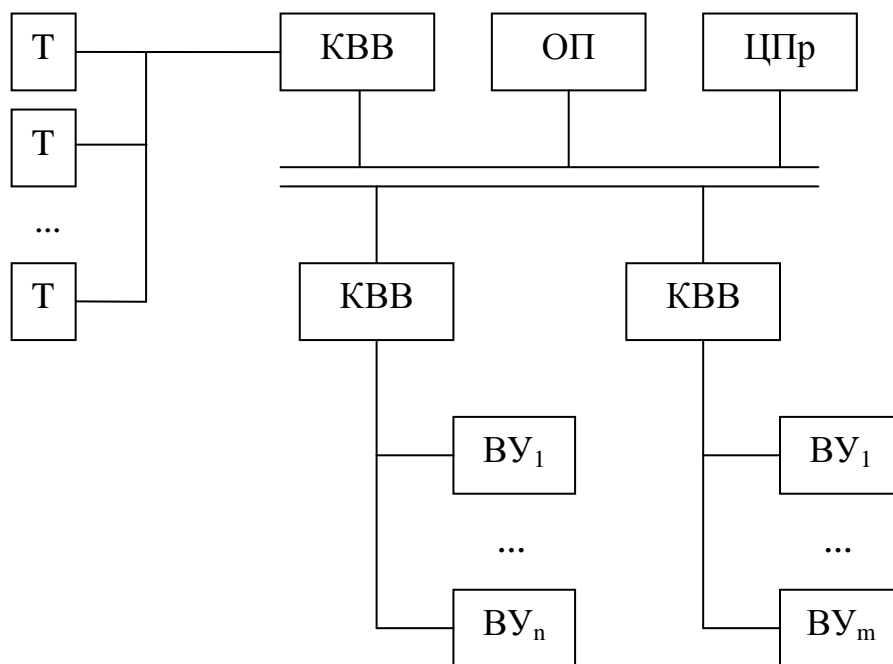


Рис. 3.1. Типовая структура системы оперативной обработки.

ЦПp – центральный процессор; *ОП* – оперативная память;
ВУ – внешние устройства; *КВВ* – контроллер ввода-вывода; *Т* – терминал

Система состоит из процессора (*ЦПp*), внешней памяти на магнитных дисках с разными характеристиками, а также множества терминалов, которые предназначены для связи с пользователями. Внешние устройства и терминалы подключаются к оперативной памяти ВС с помощью контроллеров ввода-вывода. Процесс решения задач в такой системе представляет собой последовательность этапов счета на процессоре и обращения к файлам.

При функционировании СОО в ней протекают процессы, которые разделяются на прикладные и системные. Прикладные процессы инициируются заданиями пользователей и сигналами, поступающими в нее из внешней среды. Системные процессы воспроизводят функции управления, обеспечивающие работу СОО. Они, как правило, существуют в течение всего периода работы системы – от момента ее включения до выключения.

В лабораторной работе решается задача синтеза СОО на заданном множестве устройств в двух возможных постановках

1. *С ограничением на стоимость*, которая формулируется так. Определить структуру системы, обеспечивающую минимальное среднее время ответа при заданном ограничении на стоимость системы.

2. *С ограничением на время ответа*, которая формулируется следующим образом. Определить структуру системы минимальной стоимости при ограничении на среднее время ответа.

При этом процессор считается нетиповым (вновь проектируемым) устройством, а остальные устройства — типовыми (серийно выпускаемыми промышленностью). Характеристиками типовых устройств являются быстродействие и стоимость, а нетиповых — быстродействие и стоимостной коэффициент, характеризующий зависимость стоимости от быстродействия.

При синтезе СОО определяются:

- 1) быстродействие процессора;
- 2) количество контроллеров и накопителей одного из двух типов.

В случае первой постановки задачи синтеза стоимость системы S удовлетворяет условию

$$S = \sum_{i=1}^{m_1} N_i S_i + \sum_{j=n_1+1}^n b_j V_j \leq S^*, \quad (3.1)$$

где S — стоимость системы;

S_i — стоимость стандартного устройства i -того типа;

b_j — стоимостной коэффициент нестандартного устройства j -того типа (в лабораторной работе — процессора);

V_j — его быстродействие;

S^* — ограничение на стоимость.

Для задачи синтеза во второй постановке среднее время ответа U должно удовлетворять условию

$$U = \sum_{n=1}^N \alpha_n u_n \leq U^* \quad (3.2)$$

где α_n – количество обращений задачи к n -ному устройству;

$u_n = v_n / (1 - \rho_n) = \Theta_n / (1 - \rho_n) V_n$ – среднее время пребывания задачи на этом устройстве;

U^* – допустимое время ответа (директивный срок решения задач).

Решение задачи синтеза в обеих постановках осуществляется одним и тем же методом, поэтому рассмотрим первую из них. Для синтеза системы во второй постановке достаточно в ограничениях заменить стоимость на время ответа.

Синтез системы оперативной обработки при ограничении на стоимость

Ограничение S^* на стоимость СОО при решении задачи синтеза в этой постановке выбирается из условия

$$S^* = (1,5 - 2) * S_{min}. \quad (3.3)$$

При этом получаемое значение S^* округляется до целых сотен рублей (или условных единиц), попадающих в указанный диапазон. Здесь S_{min} — стоимость минимальной конфигурации системы (содержащей минимальное количество устройств с минимальным быстродействием).

Стоимость S^* может быть распределена между перечисленными выше устройствами по-разному. При этом оптимальной считается структура, для которой среднее время ответа является минимальным; а наборы данных, используемые задачами, могут быть полностью размещены на магнитных дисках обоих типов. Количество дисков определяется из условий, обсуждаемых ниже.

Решение задачи синтеза осуществляется с использованием модели (рис. 3.2.), построенной методами теории массового обслуживания. В модели каждое устройство отображается некоторой системой массового обслуживания (СМО), состоящей из обслуживающего прибора, обозначенного квадратом, и очереди к прибору O .

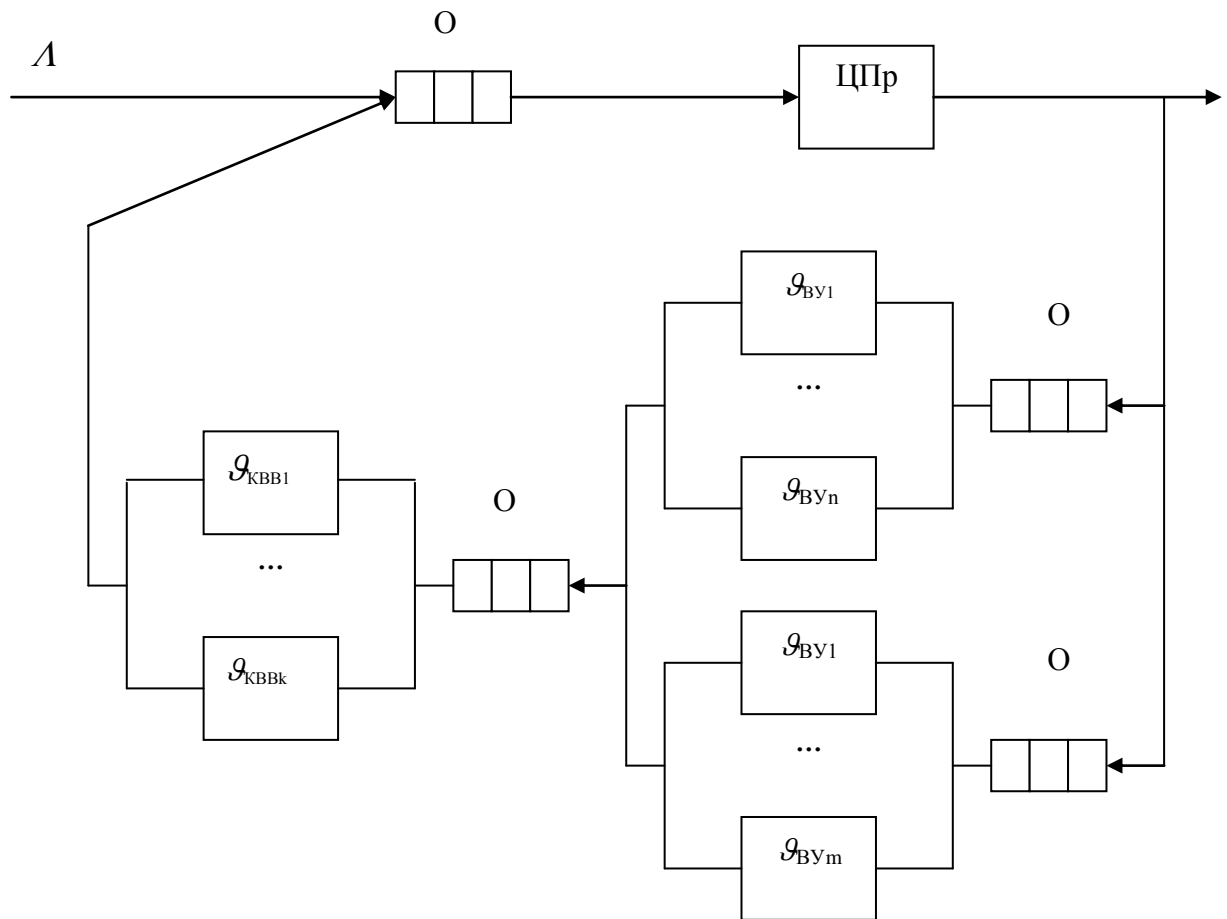


Рис. 3.2. Стохастическая сетевая модель: *ЦПр* – центральный процессор;
 BY_1, \dots, BY_n – внешние устройства типа 1;
 BY_1, \dots, BY_m – внешние устройства типа 2;

Модель СОО задается следующими параметрами:

- количеством СМО и связями между ними;
- техническими и стоимостными характеристиками устройств;
- характеристиками задач, решаемых системой.

Технические и стоимостные характеристики устройств для различных вариантов заданий приведены в табл. П.3.1 - П.3.3, а характеристики задач и их типы - в табл. П.3.4 - П.3.6.

Устройства описываются следующими параметрами:

- накопители — средним временем доступа к данным — v_1 и v_2 и емкостью накопителя G_1 и G_2 соответственно;
- контроллер — средним временем передачи данных — $v_{\text{КВВ}}$
- процессор — стоимостным коэффициентом $b_{\text{ЦПР}}$.

Определение рабочей нагрузки системы. Задачи, решаемые системой, по своим параметрам делятся на несколько классов, каждый из которых характеризуется:

- интенсивностью Λ_m поступления потока запросов на решение;
- трудоемкостью Θ (количеством) процессорных операций;
- множеством файлов F_j , используемых задачами;
- числом обращений $d_{m,j}$ к файлу F_j в процессе решения задачи m -того класса.

Файлы $F_j, j = 1, \dots, J$ описываются общей длиной файла G_j и средней длиной блока записей g_j , который передается при обмене информацией между оперативной и внешней памятью.

В модели рис. 3.2 используется однородный поток заявок на решение задач, характеристики которого могут быть получены усреднением параметров отдельных классов.

Определение параметров средней задачи осуществляется по следующим формулам.

1. Интенсивность поступления

$$\Lambda = \sum_{m=1}^M \lambda_m,$$

(3.4)

где M — количество классов задач.

2. Доля задач класса m в смеси

$$p_m = \lambda_m / \Lambda.$$

3. Трудоемкость процессорных операций для средней задачи

$$\Theta = \sum_{m=1}^M p_m \Theta_m, \quad (3.5)$$

4. Среднее число обращений к файлу F_j

$$D_j = \left(\sum_{m=1}^M \lambda_m d_{m,j} \right) / \Lambda, \quad j = \overline{1, J}. \quad (3.6)$$

5. Суммарное число обращений к файлам в процессе решения средней задачи

$$D = \sum_{j=1}^J D_j. \quad (3.7)$$

6. Средняя трудоемкость этапа счета

$$\Theta_0 = \Theta / (D + 1), \quad (3.8)$$

где $(D + 1)$ - среднее число этапов счета, приходящееся на одну среднюю задачу.

Определение параметров минимальной конфигурации СОО. В лабораторной работе минимальной считается такая структура системы оперативной обработки, в которой существует стационарный режим и все файлы размещаются в накопителях внешней памяти.

Стационарный режим в процессоре существует, если его быстродействие удовлетворяет условию

$$V_{\text{ЦПР}} > \Lambda * \Theta \quad (3.9)$$

Обычно значение, вычисленное по этой формуле, округляют в большую сторону до ближайшего числа, кратного 10.

Условие существования стационарного режима в контроллере может быть представлено в виде

$$N_{\text{КВВ}} > A * D * v_{\text{КВВ}}, \quad (3.10)$$

где $N_{\text{КВВ}}$ — количество контроллеров ввода-вывода в СОО,

$v_{\text{КВВ}}$ — среднее время передачи записей через контроллер.

Значение $N_{\text{КВВ}}$, полученное по формуле (3.10), округляется до ближайшего целого.

Стационарный режим в накопителях имеет место, если их количество $N_{\text{мд}}$ удовлетворяет условию

$$N_{\text{мд}} > A * D_{\text{мд}} * v_{\text{мд}}, \quad (3.11)$$

где $D_{\text{мд}} = D_j + D_k + \dots + D_l$ - число обращений к файлам, размещаемым на НМД каждого из двух типов. Для размещения этих файлов необходима соответствующая суммарная емкость дисковой памяти, поэтому количество указанных накопителей должно удовлетворять условию

$$N_{\text{мд}} > (G_j + G_k + \dots + G_l) / G_{\text{мд}}, \quad (3.12)$$

где G_j - длина j -того файла, $G_{\text{мд}}$ - емкость одного НМД.

Таким образом, минимальное количество НМД определяется выражением

$$N_{\text{мд}} > \text{Max}(A * D_{\text{мд}} * v_{\text{мд}}, (G_j + G_k + \dots + G_l) / G_{\text{мд}}) \quad (3.13)$$

и округляется до ближайшего целого.

Любой конфигурации СОО соответствует свое распределение файлов по типам накопителей внешней памяти. При этом стационарный режим в системе обеспечивается, если максимально допустимое время доступа $v_j^{\text{доп}}$ к файлу F_j удовлетворяет условию

$$v_j^{\text{доп}} = 1/(A * D_j) \geq v_{\text{мд}}. \quad (3.14)$$

Ограничение (3.14) используется совместно с (3.13) для определения минимальной конфигурации СОО. При этом в первую очередь размещают максимально возможное количество файлов на более медленном НМД в соответствии с условием (3.14). Остальные файлы располагают на НМД другого типа. Каждый вариант размещения характеризуется своим количеством обращений к магнитным дискам обоих типов, причем всегда должно выполняться соотношение

$$D_{\text{мд1}} + D_{\text{мд2}} = D. \quad (3.15)$$

Задача синтеза в рассматриваемой постановке считается решенной, если определена конфигурация системы, которая обеспечивает минимально возможное время ответа, а количество НМД обоих типов удовлетворяет условию (3.14). Задача синтеза во второй постановке решается аналогично. При этом ограничение на время выполнения программ выбирается по результатам синтеза СОО с ограничением на стоимость. Из них по рекомендации преподавателя выбирают одно значение времени ответа, которое принимают за допустимое время U^* .

Описание лабораторной установки

Лабораторная работа выполняется на ПЭВМ типа IBM, работающей в среде Windows 98/2000/NT и др. Программа составлена с использованием среды Delphi и позволяет сделать следующее:

1) ввести или скорректировать свою фамилию, номер варианта, а также ограничения на стоимость системы (время ответа), интенсив-

ность потока задач на ее входе и стоимостной коэффициент $b_{цпр}$ центрального процессора;

2) ввести или скорректировать параметры устройств, на которых реализуется вычислительный процесс, порождаемый задачей, и сохранить эти параметры в файле;

3) решить задачу синтеза СОО в любой постановке.

Исходные данные и результаты расчетов могут быть сохранены в файле текущего каталога на винчестере и затем распечатаны стандартными средствами.

Порядок выполнения лабораторной работы

Подготовка к работе

Подготовка предполагает выполнение следующих этапов.

1. Знакомство со всеми разделами руководства.
2. Получение у преподавателя задания на синтез СОО в предложенной постановке с конкретными характеристиками потоков задач и устройств и соответствующими ограничениями.
3. Определение параметров средней задачи по формулам (3.4)-(3.8).
4. Нахождение минимального быстродействия процессора из условия (3.9).
5. Определение минимального количества контроллеров ввода-вывода по формуле (3.10).
6. Распределение файлов по накопителям обоих типов с учетом ограничений (3.14) и определение количества обращений к этим устройствам как суммарного числа обращений задачи ко всем файлам. Общее число обращений должно удовлетворять условию (3.15).

7. Определение минимального количества НМД обоих типов из условия существования стационарного режима с использованием ограничений (3.13).

8. Вычисление стоимости минимальной конфигурации системы S_{min} по формуле (3.1).

9. Определение ограничения на стоимость S^* из условия (3.3). Значение S^* можно получить, умножив S_{min} на 2 и округлив в меньшую сторону до ближайшей сотни.

10. В лабораторной работе по указанию преподавателя должно быть исследовано от 5 до 9 вариантов структур СОО, отличающихся распределением файлов по устройствам внешней памяти. Первый вариант предполагает максимальное использование наиболее медленного типа накопителей в соответствии с ограничениями (3.14), последний — размещение файлов только на НМД более быстрого типа. В остальных вариантах исследуются промежуточные способы распределения файлов.

11. Определение ограничения на время ответа U^* по результатам расчетов, выполненных на этапе 10 настоящего раздела. Можно взять его равным времени ответа, полученному для первого варианта, и округлить это значение в сторону увеличения до десятков секунд.

12. Выполнение синтеза СОО с ограничением на время ответа U^* , используя параметры распределения файлов по накопителям разных типов, полученные на этапе 10.

13. Сравнение результатов синтеза СОО в обеих постановках.

Последовательность выполнения лабораторной работы

Программа имеет имя SOP.exe и запускается стандартным образом. Ввод исходных данных возможен из файла (если пользователь работает с программой не первый раз) или с клавиатуры в режиме диалога. По-

сле запуска приложения появляется основное окно для задания параметров задач и устройств.

Задача синтеза СОО может быть решена в обеих постановках: с ограничением на стоимость и с ограничением на время. При этом то ограничение, которое не используется в данном случае, задается равным 0. Ввод остальных параметров осуществляется обычным способом. По окончании этой операции необходимо нажать кнопку «Создать», а затем – «ОК», и программа определит и выведет количество стандартных устройств минимальной и оптимальной конфигурации, а также соответствующее быстродействие процессора.

Исходные данные и результаты расчетов могут быть сохранены в файлах, имена которых задаются пользователем с помощью меню «Файл». Они затем могут быть распечатаны обычным способом.

При выполнении лабораторной работы студенты по заданию преподавателя должны получить характеристики структур СОО для различных способов назначения файлов на устройства внешней памяти. В работе необходимо исследовать влияние такого распределения на количества устройств, получаемые в результате синтеза.

Содержание отчета о выполненной работе

Отчет должен содержать следующее.

- Название и цель работы, исходные данные.
- Характеристики средней задачи и параметры минимальной конфигурации СОО.
- Распечатки результатов, выполненных на ЭВМ.
- Модель СОО для одного из вариантов структуры (по заданию преподавателя).
- Графики зависимостей среднего времени ответа (стоимости системы) от варианта структуры или количества НМД 1 (НМД 2).

- Выводы о результатах исследования СОО и оптимальной структуре системы.

Контрольные вопросы

1. Что такое СОО и каковы особенности их функционирования?
2. Сформулируйте задачу синтеза СОО.
3. Какие параметры являются исходными для решения задачи синтеза СОО?
4. Какие параметры структуры определяются при решении задачи синтеза?
5. Что представляет собой модель СОО, применяемая для решения задачи синтеза?
6. Какие параметры задаются при решении задачи синтеза для типовых и нетиповых устройств и какие определяются в результате решения?
7. Как определить минимальное значение ограничения на стоимость системы или среднее время ее ответа, при котором постановка задачи синтеза является корректной?
8. Каким образом при синтезе учитывается способ распределения файлов между накопителями внешней памяти различного типа?
9. Какое распределение файлов является оптимальным для предложенного Вам варианта?
10. Как сохранить и распечатать исходные данные и результаты работы программы синтеза СОО?

Лабораторная работа №4

Расчет характеристик вычислительных сетей

Краткая теория

Цель работы — оценить характеристики вычислительной сети, зависимость этих характеристик от структуры сети и определить оптимальную для заданного класса задач структуру.

Типовые структуры сетей

В лабораторной работе исследуются локальные вычислительные сети и сети коллективного пользования, работающие в любом из возможных режимов: коммутации каналов, пакетов и сообщений. При этом рассматриваются физическая и логическая структура сети (состав устройств и выполняемые функции), примеры которых приведены на рис. 4.1 и 4.2. Основными элементами указанных структур являются хостмодули (абонентские машины), коммуникационные ЭВМ и каналы связи.

Физическая структура воспроизводит состав устройств и конфигурацию связей между ними. Логическая структура отображает возможные маршруты прохождения сообщений через физические каналы и узлы сети.

В лабораторной работе исследуются следующие типы физических структур (архитектуры) сетей:

- 1) петлевая (рис. 4.3);
- 2) сеть с моноканалом (линейная, рис. 4.4);
- 3) звездообразная (рис. 4.5).

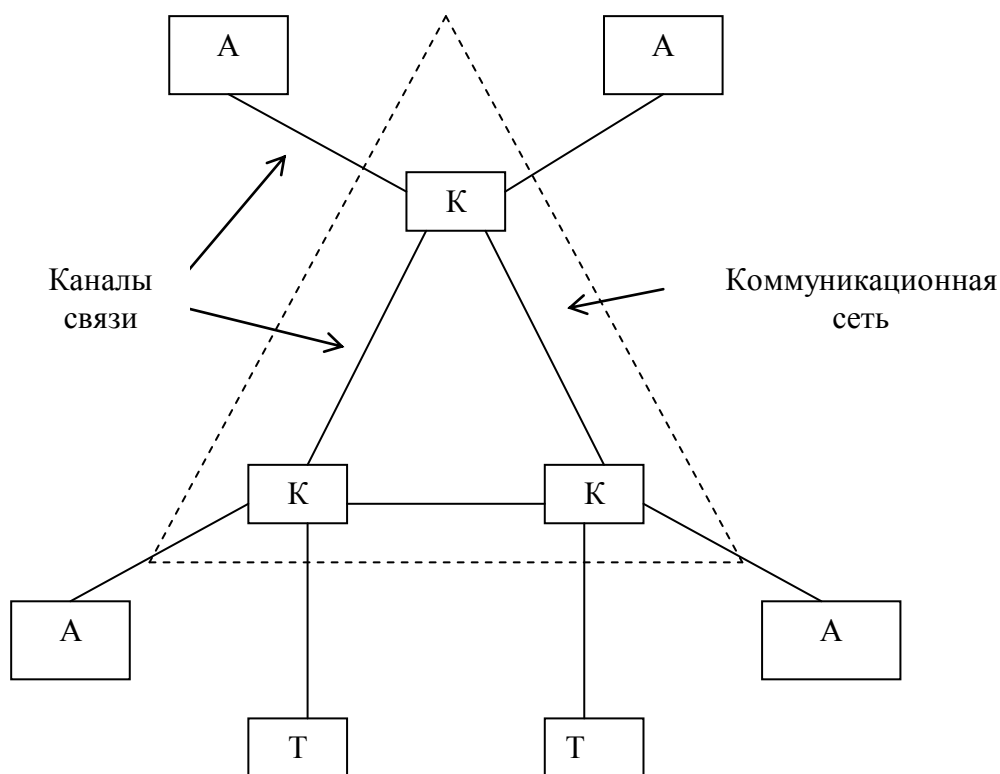


Рис. 4.1. Физическая структура вычислительной сети:
 А - абонентские машины (хостмодули); К - коммуникационные ЭВМ; Т - терминальные ЭВМ

Оценка характеристик вычислительных сетей осуществляется в лабораторной работе с использованием стохастических сетевых моделей (ССМ), в которых воспроизводится физическая структура сети и особенности передачи сообщений по каналам связи. Основные устройства сети отображаются отдельными узлами модели. Такими узлами являются системы массового обслуживания (рис. 4.6).

Система массового обслуживания (СМО) состоит из одного или нескольких обслуживающих приборов и очереди. Это – математическая модель, с помощью которой можно рассчитать временные характеристики обслуживания. На вход СМО с интенсивностью λ поступают заявки. Если прибор свободен, то он начинает обслуживание заявки. В противном случае она становится в очередь. Многоканальная СМО содержит K одинаковых приборов и позволяет одновременно

обрабатывать до K заявок.

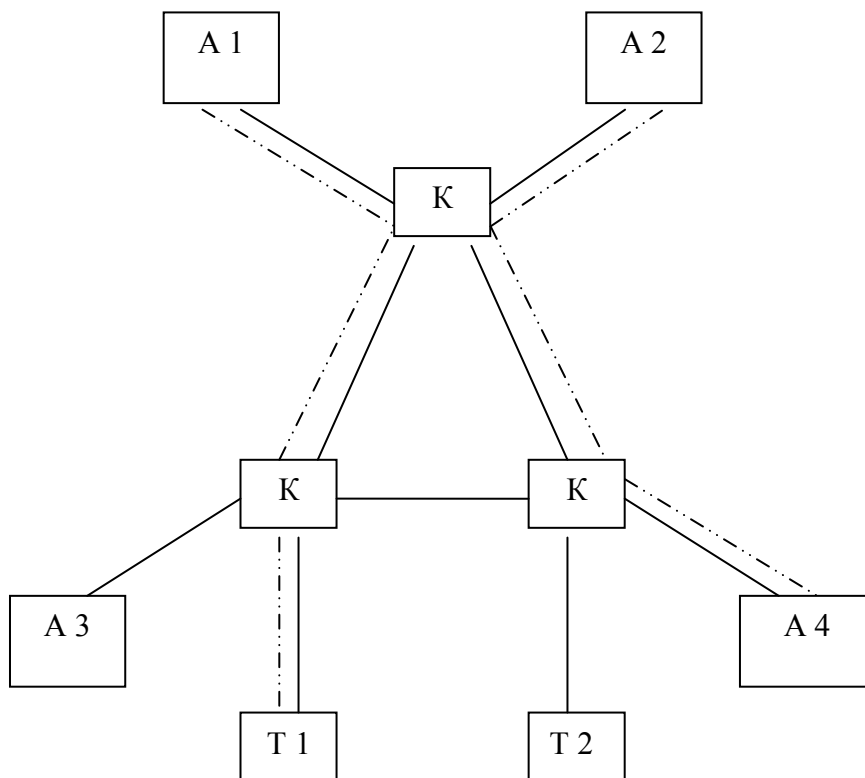


Рис. 4.2. Логическая структура вычислительной сети:
Штрих пунктиром изображены каналы между
хостмодулями А2 и А4 и ЭВМ А1 и Т1

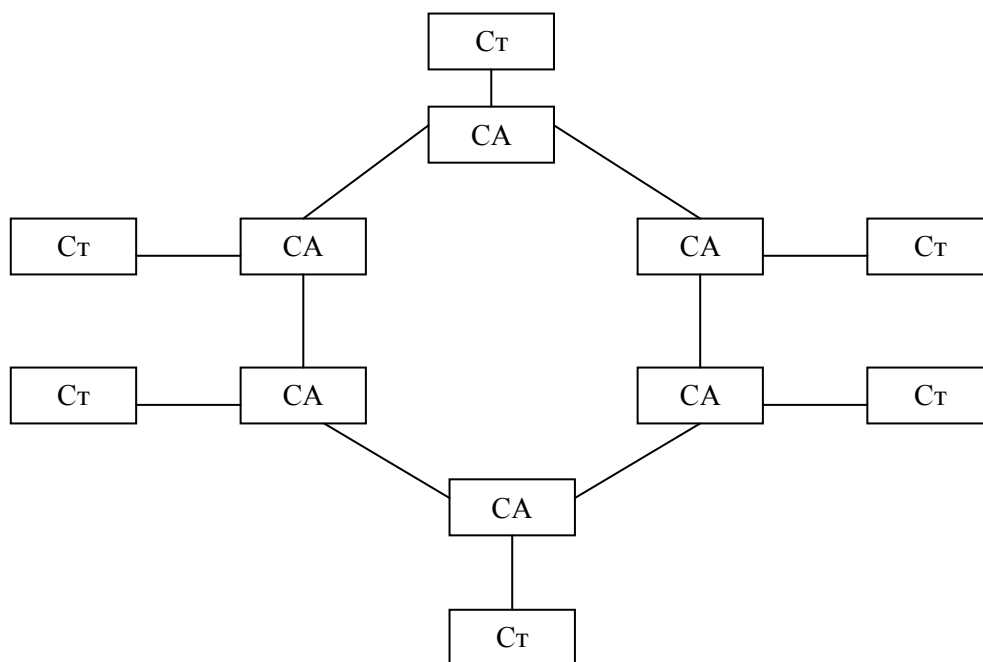


Рис. 4.3. Вычислительная сеть с петлевой структурой:
Ст - станция. СА - сетевой адаптер

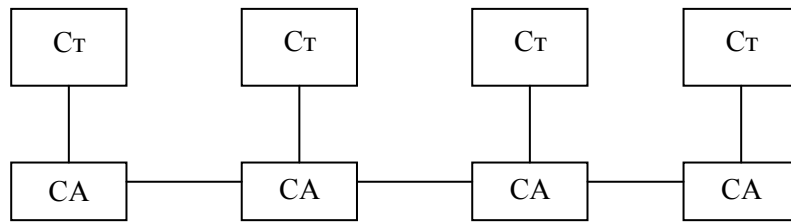


Рис. 4.4. Вычислительная сеть линейной структуры:
Ст - станция, СА - сетевой адаптер

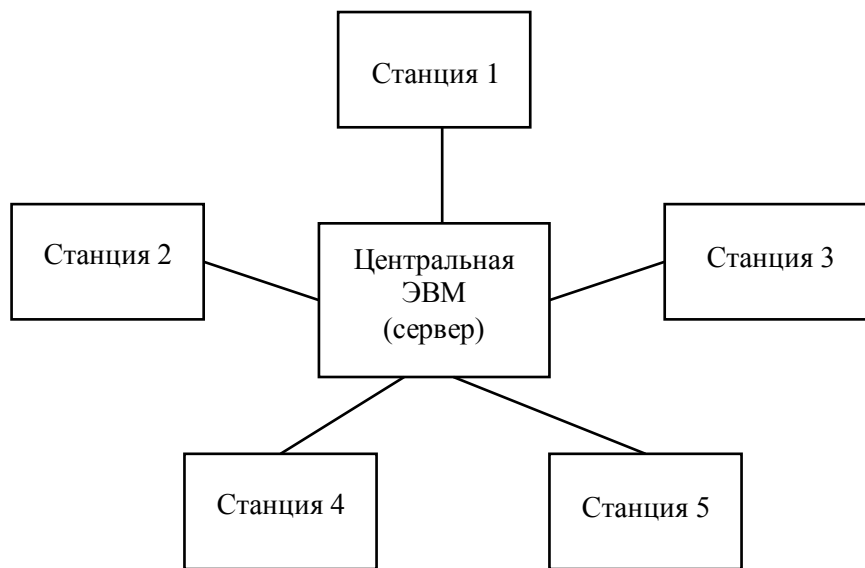


Рис. 4.5. Вычислительная сеть звездообразной структуры

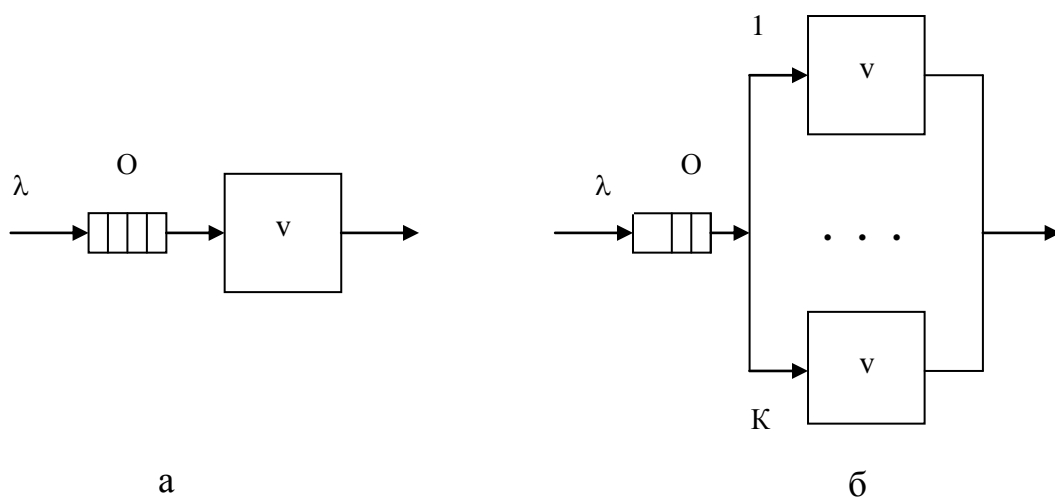


Рис. 4.6. Модель устройств сети - система массового обслуживания (СМО):

а - одноканальная, б - многоканальная;

λ - интенсивность потока заявок на обслуживание; О - очередь;
v - среднее время обслуживания; К - количество каналов в СМО

Моделью вычислительной сети в целом является совокупность взаимосвязанных СМО. Она, как отмечалось выше, называется стохастической сетью. В модели отображаются только те устройства, которые вносят задержку в вычислительный процесс, а именно, абонентские ЭВМ и каналы связи. Терминалы, адаптеры и другие устройства обычно не представляются. Возможные маршруты прохождения пакетов по вычислительной сети задаются связями между соответствующими СМО, которые представляются в форме матрицы вероятностей передач. Источник запросов на решение задач воспроизводится в модели некоторой фиктивной СМО S_0 . При этом модель в целом является разомкнутой стохастической сетью (сетью массового обслуживания).

Возможные варианты моделей петлевой, линейной и звездообразной структур вычислительных сетей приведены на рис. 4.7 – 4.9.

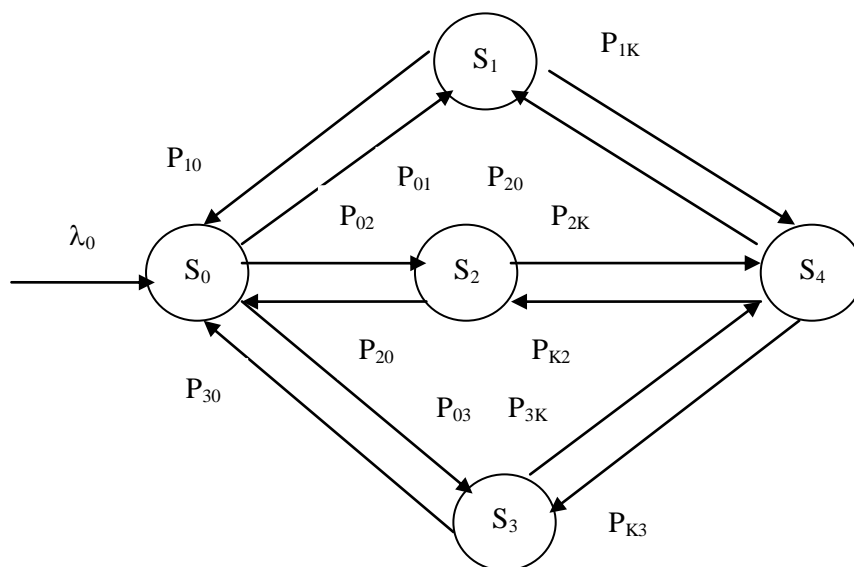


Рис. 4.7. Модель вычислительной сети с петлевой структурой:

S_0 - источник заявок (задач); S_1 - S_3 - абонентские ЭВМ;

λ_0 - интенсивность входного потока заявок

Параметрами (исходными данными для расчета) модели являются:

- 1) количество N СМО (устройств), которые в ней отображаются;

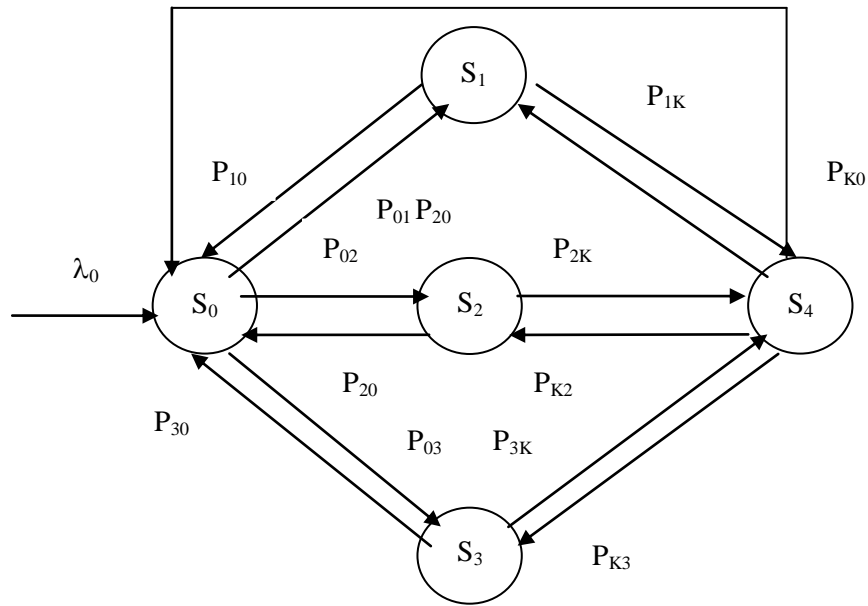


Рис. 4.8. Модель вычислительной сети с линейной структурой: S_0 - источник заявок; S_1 - S_3 - абонентские ЭВМ; λ_0 - интенсивность входного потока заявок

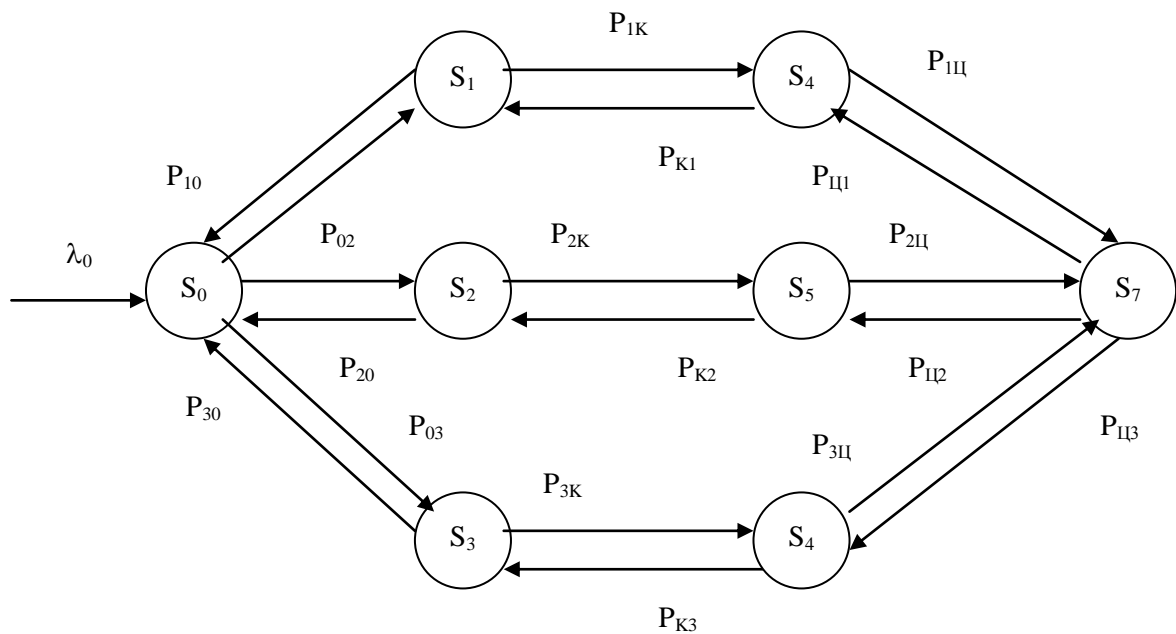


Рис. 4.9. Модель вычислительной сети со звездообразной структурой. S_0 - источник заявок (задачи); S_1 - S_3 - абонентские ЭВМ; S_4 - S_6 - каналы связи; S_7 - абонентская ЭВМ; λ_0 - интенсивность входного потока заявок

2) тип каждой системы массового обслуживания (количество каналов в ней);

3) матрица вероятностей передач $P = [p_{i,j}]$, $i, j = 0, \dots, N$, элемент $p_{i,j}$ которой представляет собой вероятность того, что задача (заявка) после устройства i будет использовать устройство j (перейдет из СМО i в СМО j). Матрица P является стохастической, поэтому сумма вероятностей в любой ее строке должна быть равна 1, а в столбцах должен быть хотя бы один ненулевой элемент;

4) интенсивность λ_0 поступления заявок на решение задач сетью;

5) средние длительности использования задачами отдельных ресурсов (устройств) сети (трудоемкости обслуживания заявок в соответствующих СМО).

Основные параметры определяются типом модели вычислительной сети и задаются преподавателем в соответствии с характеристиками классов решаемых задач, перечисленными в прил. 4.

Для петлевой структуры вероятности обращения абонентских ЭВМ к каналам вычисляются по формуле

$$p_{i,k} = \sum_{\substack{j=1, \\ j \neq i}}^M n_{i,j} / \sum_{j=1}^M n_{i,j}, i = \overline{1, M} \quad (4.1)$$

где M – число абонентских ЭВМ сети;

$n_{i,j}$ - количество пакетов сообщений, передаваемых i -той абонентской машиной (задаются строкой табл. П.4.3);

$n_{i,i}$ - количество пакетов, обрабатываемых данной (i -той) ЭВМ.

Вероятности решения задач непосредственно на i -той абонентской ЭВМ определяются количеством пакетов, которые эта машина передает самой себе. Они соответствуют переходам из СМО S_i в СМО S_0 в модели и определяются так

$$p_{i,0} = n_{i,i} / \sum_{j=1}^M n_{i,j}, i = \overline{1, M} \quad (4.2)$$

Вероятности передач сообщений из канала в i -тую ЭВМ вычисляются следующим образом

$$p_{k,i} = n_{k,i} / \sum_{i=1}^M \sum_{\substack{j=1, \\ j \neq i}}^M n_{i,j}, i = \overline{1, M} \quad (4.3)$$

где $n_{k,i} = \sum_{\substack{j=1, \\ j \neq i}}^M n_{i,j}$ — количество пакетов, получаемых i -той абонентской ЭВМ, задаются столбцом (см. табл. П.4.3).

$\sum_{i=1}^M \sum_{\substack{j=1, \\ j \neq i}}^M n_{i,j}$ — общее количество пакетов, циркулирующих в канале.

Например, для задач класса 1 из табл. П.4.3 получаем следующие значения.

1. Количество обращений абонентских ЭВМ к каналу:

$$n_{1K} = n_{12} + n_{13} = 10 + 14 = 24;$$

$$n_{2K} = n_{21} + n_{23} = 30;$$

$$n_{3K} = n_{31} + n_{32} = 30.$$

2. Вероятности обращений к каналу и решения задач на данной ЭВМ:

$$p_{1K} = n_{1K} / (n_{1K} + n_{11}) = 24/40 = 0.6;$$

$$p_{10} = 16/40 = 0.4;$$

$$p_{2K} = n_{2K} / (n_{2K} + n_{22}) = 30/50 = 0.6;$$

$$p_{20} = 20/50 = 0.4;$$

$$p_{3K} = n_{3K} / (n_{3K} + n_{33}) = 30/40 = 0.75;$$

$$p_{30} = 10/40 = 0.25.$$

3. Количество передач пакетов из канала в абонентские ЭВМ

$$n_{K1} = n_{21} + n_{31} = 5 + 20 = 25;$$

$$n_{K2} = n_{12} + n_{32} = 20;$$

$$n_{K3} = n_{13} + n_{23} = 39.$$

4. Вероятности передач пакетов из канала в абонентские ЭВМ

$$p_{K1} = n_{K1} / (n_{21} + n_{31} + n_{12} + n_{32} + n_{13} + n_{23}) = 25/84 \approx 0.3;$$

$$p_{K2} = n_{K2} / (n_{21} + n_{31} + n_{12} + n_{32} + n_{13} + n_{23}) = 20/84 \approx 0.24;$$

$$p_{K3} = n_{K3} / (n_{21} + n_{31} + n_{12} + n_{32} + n_{13} + n_{23}) = 39/84 \approx 0.46.$$

5. Матрица вероятностей передач для задач класса 1 из табл. 3 прил. 4 при петлевой архитектуре сети имеет вид

$$P = \begin{matrix} & \begin{matrix} S_0 & S_1 & S_2 & S_3 & S_4 \end{matrix} \\ \begin{matrix} S_0 \\ S_1 \\ S_2 \\ S_3 \\ S_4 \end{matrix} & \begin{vmatrix} 0 & 0.1 & 0.4 & 0.5 & 0 \\ 0.4 & 0 & 0 & 0 & 0.6 \\ 0.4 & 0 & 0 & 0 & 0.6 \\ 0.25 & 0 & 0 & 0 & 0.75 \\ 0 & 0.3 & 0.24 & 0.46 & 0 \end{vmatrix} \end{matrix}$$

Для линейной структуры необходимо учесть, что сообщения от каждой абонентской ЭВМ передаются по каналу в обоих направлениях, поэтому при определении вероятностей p_{iK} в формулу (4.1) необходимо подставить удвоенные значения количеств передаваемых пакетов.

$$n_{i,K}^n = 2n_{i,K}, \quad i = 1, 2, 3,$$

Таким образом, число пакетов, отправляемых в канал i -той ЭВМ определяется из соотношения

где $n_{i,K}$ - количество обращений к каналу i -той ЭВМ для петлевой сети.

Вероятности p_{i0} вычисляются по тем же формулам, что и для петли. Вероятности p_{Ki} передач пакетов из канала в абонентские ЭВМ определяются в соответствии с выражением

$$p_{K,i} = n_{K,i} / 2 \sum_{i=1}^M \sum_{\substack{j=1, \\ j \neq i}}^M n_{i,j}, \quad i = 1, \dots, M, \quad (4.4)$$

где $2 \sum_{i=1}^M \sum_{\substack{j=1, \\ j \neq i}}^M n_{i,j}$ — общее количество пакетов, циркулирующих в канале.

В линейных структурах вводится дополнительная операция — удаление пакетов из канала, которая вызвана тем, что только половина отправленных каждой станцией пакетов доходит до адресатов. Другая половина попадает в терминаторы, расположенные на концах линии связи, и поглощается ими. В модели эта операция отображается переходом из СМО S_K в СМО S_0 с вероятностью

$$p_{i0} = 0.5, \quad i = 1, 2, 3.$$

Для задачи класса 1 из табл. П.4.3 получаем следующие параметры.

1. Количество обращений абонентских ЭВМ к каналу

$$n_{1K} = 2n_{12} + 2n_{13} = 20 + 28 = 48;$$

$$n_{2K} = 2n_{21} + 2n_{23} = 60;$$

$$n_{3K} = 2n_{31} + 2n_{32} = 60.$$

2. Вероятности обращений к каналу и решения задач на данной ЭВМ:

$$p_{1K} = n_{1K} / (n_{1K} + n_{11}) = 48 / (48 + 16) = 0.75; \quad p_{10} = 16 / 64 = 0.25;$$

$$p_{2K} = n_{2K} / (n_{2K} + n_{22}) = 60 / (60 + 20) = 0.75; \quad p_{20} = 20 / 80 = 0.25;$$

$$p_{3K} = n_{3K} / (n_{3K} + n_{33}) = 60 / (60 + 10) = 0.89; \quad p_{30} = 10 / 70 = 0.11.$$

3. Вероятности передач пакетов из канала в абонентские ЭВМ

$$p_{K1} = (n_{21} + n_{31}) / 2(n_{21} + n_{31} + n_{12} + n_{32} + n_{13} + n_{23}) = 25 / 168 \approx 0.15;$$

$$p_{K2} = (n_{12} + n_{32}) / 2(n_{21} + n_{31} + n_{12} + n_{32} + n_{13} + n_{23}) = 20 / 168 \approx 0.12;$$

$$p_{K3} = (n_{13} + n_{23}) / 2(n_{21} + n_{31} + n_{12} + n_{32} + n_{13} + n_{23}) = 39 / 168 \approx 0.23;$$

$$p_{i0} = 0.5, \quad i = 1, 2, 3.$$

Матрица вероятностей передач для задач класса 1 при линейной архитектуре имеет вид

$$P = \begin{array}{c} S_0 \\ S_1 \\ S_2 \\ S_3 \\ S_4 \end{array} \left| \begin{array}{ccccc} S_0 & S_1 & S_2 & S_3 & S_4 \\ 0 & 0.1 & 0.4 & 0.5 & 0 \\ 0.25 & 0 & 0 & 0 & 0.75 \\ 0.25 & 0 & 0 & 0 & 0.75 \\ 0.11 & 0 & 0 & 0 & 0.89 \\ 0.5 & 0.15 & 0.12 & 0.23 & 0 \end{array} \right|$$

Для звездообразной структуры вероятности передач пакетов сообщений в каналы определяются по тем же формулам, что и у петли. При этом можно считать, что пакеты двигаются по каналу в обоих направлениях (от абонентских ЭВМ к центральной и обратно) с равной вероятностью, т.е.

$$p_{iK} = p_{Ki} = 0.5.$$

$$p_{Ц,j} = n_{K,j} / \sum_{i=1}^M n_{k,i}, j = \overline{1, M}.$$

Центральная ЭВМ передает пакеты в j -тый канал с вероятностью

Таким образом, матрица вероятностей передач для сети типа "звезда" будет подобна матрице сети с петлевой архитектурой (только растянутой на большее количество узлов). Для задач класса 1 (см. прил. 4) она будет иметь следующий вид

$$P = \begin{array}{c} S_0 \\ S_1 \\ S_2 \\ S_3 \\ S_4 \\ S_5 \\ S_6 \\ S_7 \end{array} \left| \begin{array}{cccccccc} S_0 & S_1 & S_2 & S_3 & S_4 & S_5 & S_6 & S_7 \\ 0 & 0.1 & 0.4 & 0.5 & 0 & 0 & 0 & 0 \\ 0.4 & 0 & 0 & 0 & 0.6 & 0 & 0 & 0 \\ 0.4 & 0 & 0 & 0 & 0 & 0.6 & 0 & 0 \\ 0.25 & 0 & 0 & 0 & 0 & 0 & 0.75 & 0 \\ 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0.5 \\ 0 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0.5 \\ 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0.5 \\ 0 & 0 & 0 & 0 & 0.3 & 0.24 & 0.46 & 0 \end{array} \right|$$

Описание лабораторной установки

Лабораторная работа выполняется на ПЭВМ типа IBM, работающей в среде Windows 98/2000/NT и др. Программа составлена с использованием среды Delphi. Приложение имеет имя Pvs v 2.0 и позволяет сделать следующее.

1. Ввести интенсивность входного потока.
2. Задать количество узлов (смо) сетевой модели.
3. Ввести параметры каждого узла:
 - количество каналов;
 - средние времена обслуживания.
4. Задать связи между узлами с помощью матрицы вероятностей передач.

Программа обеспечивает оценку следующих характеристик сетевой модели и соответствующей ей вычислительной сети:

1. Для каждого узла (устройства) сети:
 - а) коэффициент загрузки;
 - б) среднее количество заявок в узле и очереди на обслуживание;
 - в) средние времена ожидания и пребывания заявок.
2. Для сети в целом:
 - а) среднее количество заявок (задач) в сети (уровень мультипрограммирования);
 - б) средние времена ожидания и пребывания заявок (время ответа);
 - в) среднюю длину очереди.

Перечисленные характеристики позволяют для каждого из четырех типов архитектур вычислительной сети выявить так называемые «узкие места» (наиболее загруженные участки сети) и дать рекомендации

о построении сбалансированной структуры, у которой все устройства загружены одинаково. При этом основной характеристикой служит коэффициент загрузки каждого узла. Если он больше 0.5, то соответствующее устройство можно считать «узким местом». Разгрузка такого узла достигается за счет добавления устройств (каналов в СМО).

Результаты расчетов выводятся на экран дисплея или в файл, имя которого задает пользователь. Информация из файла может быть затем отпечатана обычным образом.

Порядок выполнения лабораторной работы

Подготовка к работе

Подготовка предполагает выполнение следующих этапов.

1. Знакомство со всеми разделами руководства.
2. Получение у преподавателя варианта задания на исследование сети с конкретным классом задач.
3. Построение моделей вычислительных сетей каждого из трех типов структур, рассмотренных в теоретической части.
4. Расчет параметров матрицы P для модели каждого типа в соответствии с выражениями (4.1)—(4.9).

Последовательность выполнения лабораторной работы

Программа, как отмечалось выше, имеет имя Pvs v 2.0 и запускается стандартным образом. Ввод исходных данных возможен из файла (если пользователь работает с ней не первый раз) или с клавиатуры в режиме диалога. После запуска программы появляется основное окно для задания интенсивности поступления заявок, параметров узлов мо-

дели и матрицы вероятностей передач. Ввод сопровождается контролем исходных данных. В частности, **матрица P является стохастической, поэтому сумма вероятностей в любой ее строке должна быть равна 1, а в столбцах должен быть хотя бы один ненулевой элемент.**

Исходные данные могут быть загружены из файла. Для этого необходимо выбрать соответствующий пункт в меню "Файл". Сохранение результатов выполняется аналогично.

Расчет характеристик производится после нажатия кнопки "Начать". Если исходные данные заданы неверно, то может появиться сообщение "Система ... перегружена". В этом случае необходимо проверить значения параметров соответствующей СМО.

Все результаты расчетов рекомендуется сохранять и по возможности распечатать, чтобы включить в отчет.

Содержание отчета о выполненной работе

Отчет должен содержать следующее:

- название и цель работы, исходные данные;
- модели исследованных структур сетей;
- результаты расчетов;
- выводы о том, какая архитектура сети является лучшей, а какая — худшей.

Контрольные вопросы

1. Какие типы вычислительных сетей Вы знаете?
2. Какие типы архитектур сетей Вам известны?
3. Какие модели используются для исследования вычислительных сетей?
4. Какими элементами модели воспроизводятся особенности работы отдельных узлов сети?
5. Что является параметрами модели вычислительной сети?

6. Что является характеристиками модели сети?
7. Как определяются элементы матрицы вероятностей передач для моделей сетей различной структуры?
9. Как изменяются характеристики сети при изменении количества каналов наиболее загруженного узла?
10. Каковы Ваши предложения по оптимизации структуры исследованных Вами сетей?

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Основы теории вычислительных систем /Под ред. Майорова С. А. - М.: Высш. шк., 1978. 408 с.
2. Майоров С.А., Новиков Г.И. Структура электронных вычислительных машин. - Л.: Машиностроение, 1979. 384 с.
3. Методические указания к лабораторным работам по курсу "Вычислительные комплексы, системы и сети" для студентов специальности 0608 / Куйбыш. Политехн. ин-т; Сост. *Н.В.Ефимушкина*. Куйбышев, 1988. 28 с.
3. Методические указания к лабораторным работам "Расчет характеристик вычислительных сетей и систем пакетной обработки" по курсу "Вычислительные комплексы, системы и сети" для студентов специальности 0608 / Куйбыш. Политехн. ин-т; Сост. *Н.В.Ефимушкина*. Куйбышев, 1988. 28 с.

***Типы вычислительных процессов,
исследуемых в лабораторной работе 1***

1. Ввод и отладка на станции локальной сети программы, записанной на языке высокого уровня.
2. Ввод и отладка на ПЭВМ программы, записанной на языке высокого уровня.
3. Редактирование текста на ПЭВМ. Вывод выходных документов на печать.
4. Работа на станции локальной сети с базой данных, находящейся на сервере.
5. Работа на станции локальной сети с базой данных, находящейся на этой станции.
6. Работа с готовой программой на станции локальной сети с использованием файлов, находящихся на сервере.
7. Работа с готовой программой на станции локальной сети с использованием файлов, находящихся на этой станции.
8. Работа в локальной сети в режиме телеконференции (количество участников телеконференции — не менее 4).
9. Работа на станции локальной сети с базой данных, находящейся на сервере. Вывод выходных документов на печать.
10. Работа на станции локальной сети с базой данных, находящейся на этой станции. Вывод выходных документов на печать.

Основные исходные данные для оценки трудоемкости алгоритмов

Вариант	Количество строк (записей)				Количество операций процессора при выполнении программы, млн		Вероятности ошибок	
	При вводе информации		При выводе информации					
	Минимум	Максимум	Минимум	Максимум	Минимум	Максимум	Минимум	Максимум
1	20	1000			1	100	0.01	0.5
2	20	1000			1	100	0.01	0.5
3	20	1000	20	1000	0.1	10	0.01	0.5
4	10	100	10	100	0.1	10	0.01	0.5
5	10	100	10	100	0.1	10	0.01	0.5
6	20	1000	20	1000	0.1	10	0.01	0.5
7	20	1000	20	1000	0.1	10	0.01	0.5
8	5	10	5	10	0.01	1	0.01	0.5
9	20	1000	20	1000	0.1	10	0.01	0.5
10	20	1000	20	1000	0.1	10	0.01	0.5

Таблица П.2.1.

Варианты заданий

Номер варианта	Номера потоков
1	1, 5, 9, 13, 17, 21
2	2, 6, 10, 15, 20, 30
3	29, 25, 23, 19, 15, 11
4	4, 7, 10, 13, 16, 19
5	30, 23, 16, 9, 2, 26
6	5, 10, 15, 20, 25, 30
7	10, 9, 8, 7, 3, 4
8	8, 12, 16, 20, 24, 28
9	11, 12, 13, 14, 29, 5
10	12, 16, 8, 4, 22, 25
11	13, 18, 28, 30, 2, 7
12	14, 17, 29, 11, 8, 9
13	15, 16, 9, 8, 2, 21
14	30, 26, 22, 18, 14, 10
15	1, 8, 12, 20, 24, 30
16	3, 6, 9, 12, 15, 18
17	1, 6, 11, 16, 21, 26
18	6, 12, 18, 24, 30, 1
19	7, 13, 19, 26, 30, 2
20	23, 22, 21, 20, 19, 18
21	9, 5, 1, 16, 20, 24
22	28, 23, 18, 13, 8, 29
23	20, 23, 12, 15, 5, 6
24	21, 23, 25, 29, 8
25	22, 3, 16, 19, 1, 23
26	27, 28, 17, 13, 5, 20
27	10, 2, 8, 17, 4, 25
28	11, 1, 9, 18, 3, 26
29	12, 3, 15, 19, 5, 27
30	13, 4, 16, 20, 6, 28

Характеристики потоков заявок

Номер потока	Интенсивность поступления, с	Трудоемкость обслуживания, тыс. операций	Коэффициент вариации трудоемкости	Допустимое среднее время пребывания, с
1	0.1	400	0	5.0
2	0.4	250	1	10.0
3	1.8	105	1	6.7
4	5.9	25	0	4.6
5	2.6	50	0	5.1
6	3.4	40	1	6.3
7	0.8	240	1	7.2
8	0.3	320	0	6.1
9	1.5	150	0	5.3
10	4.1	70	1	8.4
11	0.7	500	0	10.0
12	3.5	65	0	3.8
13	0.25	450	1	7.7
14	0.66	180	1	5.5
15	0.93	240	0	10.2
16	1.3	100	0	2.8
17	2.1	80	1	5.3
18	0.5	200	1	3.3
19	0.6	250	0	4.8
20	2.5	180	0	5.2
21	3.7	90	1	4.3
22	1.2	60	1	1.8
23	0.85	150	0	2.6
24	1.1	220	0	3.1
25	1.9	280	1	5.7
26	2.0	120	1	3.6
27	0.75	300	0	6.8
28	3.3	110	0	4.9
29	0.36	400	1	1.5
30	4.3	40	1	5.6

Таблица П.3.1.

Параметры накопителей

Номер варианта	Среднее время доступа к накопителю $v_{мд}$, с		Емкость накопителя $G_{мд}$, Мбайт	
	Тип 1	Тип 2	Тип 1	Тип 2
1	1.0	0.09	2000	500
2	1.2	0.08	1800	400
3	1.3	0.07	1600	300
4	1.4	0.06	1400	300
5	1.5	0.05	1200	100
6	1.4	0.04	1000	100
7	1.3	0.03	1200	200
8	1.2	0.02	1400	300
9	1.0	0.01	1600	400
10	1.2	0.02	1800	500

Таблица П3.2.

Параметры контроллера

Номер варианта	1	2	3	4	5	6	7	8	9	10
Среднее время передачи, с	0.1	0.15	0.2	0.25	0.2	0.15	0.1	0.15	0.2	0.25

Таблица П.3.3.

Стоимость устройств

№ варианта	1	2	3	4	5	6	7	8	9	10	
Стоимостной коэффициент процессора	1	1.5	2	2.5	3	3.5	4	4.5	5	5.5	
Накопители	типа 1	20	25	30	35	40	50	40	35	30	25
	типа 2	100	130	120	110	100	120	150	160	180	120
Контроллер	50	60	70	80	70	60	50	60	70	80	

Таблица П.3.4.

Параметры задач, решаемых системой

№ варианта	Задача 1		Задача 2		Задача 3		Задача 4		Задача 5	
	№ задачи	Интенсивность	№ задачи	Интенсивность	№ задачи	Интенсивность	№ задачи	Интенсивность	№ задачи	Интенсивность
1	1	0.10	20	0.05	5	0,01	16	0,08	10	0,03
2	2	0.09	19	0.06	4	0,02	15	0,07	11	0,04
3	3	0.08	18	0.07	2	0,03	14	0,06	12	0,05
4	4	0.07	17	0.08	3	0,04	13	0,05	14	0,06
5	5	0.06	16	0.09	1	0,05	12	0,04	13	0,07
6	6	0.05	15	0.10	1	0,06	11	0,03	16	0,08
7	7	0.04	14	0.10	2	0,07	10	0,02	15	0,09
8	8	0.03	13	0.09	3	0,08	9	0,01	17	0,10
9	9	0.02	12	0.08	4	0,09	8	0,01	18	0,10
10	10	0.01	11	0.07	5	0,10	7	0,02	19	0,09
11	11	0.01	10	0.06	6	0,10	5	0,02	20	0,08
12	12	0.02	9	0.05	7	0,09	6	0,03	20	0,08
13	13	0.03	8	0.04	20	0,08	4	0,04	19	0,07
14	14	0.04	7	0.03	9	0,07	3	0,05	18	0,06
15	15	0.05	6	0.02	10	0,06	2	0,06	17	0,05
16	16	0.06	5	0.01	11	0,05	1	0,07	15	0,05
17	17	0.07	4	0.01	12	0,04	1	0,08	16	0,04
18	18	0.08	3	0.02	13	0,03	2	0,09	14	0,03
19	19	0.09	2	0.03	14	0,02	3	0,10	13	0,02
20	20	0.10	1	0.04	15	0,01	4	0,10	12	0,01
21	19	0.10	1	0.05	16	0,01	5	0,09	11	0,02
22	18	0.09	2	0.06	17	0,01	6	0,08	10	0,03
23	17	0.08	3	0.07	18	0,02	7	0,07	9	0,04
24	16	0.07	4	0.08	19	0,03	8	0,06	8	0,05
25	15	0.06	5	0.09	20	0,04	9	0,05	7	0,06

Таблица П.3.5

Параметры файлов

	Номер файла									
	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
Длина файла, Мбайт	120	100	180	150	180	100	120	150	180	110

Таблица П.3.6

Трудоемкости задач и число обращений к файлам

Номер задачи	Трудоемкость млн операций	Число обращений к файлам									
		F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
1	1000	5			3			4		6	
2	900		4			5			3		2
3	800			3			4		5		10
4	700	6			2			10		3	
5	600		5				3		2		6
6	500			4		6				4	
7	400				4		6		4		1
8	300	4			3			10		5	
9	200		6			3			2		5
10	100			5		4			10		2
11	100	5			5			2		4	
12	200		6				2		3		4
13	300	10		6				4	2		3
14	400		2		6		3			4	
15	500	3				5			10		6
16	600		5					6		3	4
17	700			6			2		3		10
18	800	6			2			2		4	
19	900		4			6			4		10
20	1000			5			3		3		2

Таблица П.4.1.

Трудоемкости обслуживания задач и пакетов устройствами сети

№ класса задач	Трудоемкости обслуживания, с.				
	Абонентскими ЭВМ			Центральной ЭВМ	Каналами связи
	1	2	3		
1.	0.2	0.1	0.25	0.01	0.002
2.	0.25	0.15	0.1	0.02	0.003
3.	0.05	0.2	0.15	0.015	0.0025
4.	0.08	0.1	0.25	0.025	0.002
5.	0.2	0.08	0.16	0.02	0.0015
6.	0.4	0.3	0.05	0.015	0.001
7.	0.35	0.2	0.08	0.012	0.0015
8.	0.15	0.1	0.3	0.01	0.002
9.	0.1	0.15	0.28	0.012	0.0025
10.	0.15	0.12	0.4	0.015	0.003
11.	0.2	0.25	0.05	0.02	0.0025
12.	0.3	0.1	0.15	0.025	0.002
13.	0.35	0.2	0.08	0.02	0.0015
14.	0.4	0.16	0.05	0.015	0.001
15.	0.3	0.25	0.1	0.012	0.0015
16.	0.1	0.3	0.16	0.01	0.002
17.	0.25	0.1	0.3	0.012	0.0025
18.	0.18	0.28	0.1	0.015	0.002
19.	0.2	0.15	0.1	0.02	0.0015
20.	0.1	0.2	0.3	0.025	0.001

Таблица П.4.2.

Интенсивности поступления задач и вероятности обращения к абонентским ЭВМ сети

№ класса задач	Интенсивность поступления, с ⁻¹	Вероятности обращения к абонентским ЭВМ		
		1	2	3
1	1.0	0.1	0.4	0.5
2	1.5	0.8	0.05	0.15
3	2.4	0.35	0.2	0.45
4	2.0	0.45	0.35	0.2
5	1.5	0.5	0.2	0.3
6	1.0	0.2	0.55	0.25
7	1.5	0.15	0.6	0.25
8	2.0	0.3	0.25	0.45
9	2.4	0.75	0.1	0.15
10	2.2	0.65	0.25	0.1
11	2.0	0.55	0.1	0.35
12	1.8	0.25	0.35	0.4
13	1.6	0.4	0.5	0.1
14	1.5	0.2	0.3	0.5
15	1.2	0.2	0.1	0.7
16	1.0	0.1	0.1	0.8
17	1.2	0.05	0.8	0.15
18	1.5	0.3	0.4	0.3
19	1.8	0.45	0.45	0.1
20	2.0	0.35	0.1	0.55

Таблица П.4.3.

Средние количества пакетов, передаваемых абонентским ЭВМ сети

№ класса задач	№ абонентской ЭВМ				№ класса задач	№ абонентской ЭВМ			
		1	2	3			1	2	3
1	1	16	10	14	11	1	30	10	10
	2	5	20	25		2	10	20	20
	3	20	10	10		3	8	16	26
2	1	8	22	10	12	1	14	26	10
	2	25	10	15		2	10	25	15
	3	10	25	15		3	10	10	30
3	1	20	4	16	13	1	8	26	16
	2	15	20	10		2	25	15	10
	3	25	10	15		3	10	30	10
4	1	10	25	15	14	1	26	6	18
	2	22	18	10		2	8	30	12
	3	15	10	25		3	16	14	20
5	1	25	10	15	15	1	12	20	18
	2	12	20	18		2	25	10	15
	3	25	10	15		3	16	24	10
6	1	18	12	20	16	1	5	4	11
	2	8	22	10		2	7	18	15
	3	5	20	25		3	28	12	10
7	1	14	16	20	17	1	32	8	10
	2	20	5	25		2	17	18	15
	3	20	4	16		3	25	11	14
8	1	25	10	15	18	1	16	24	10
	2	18	12	20		2	20	15	15
	3	5	15	25		3	10	15	25
9	1	8	12	20	19	1	15	7	18
	2	16	20	14		2	25	15	10
	3	25	15	10		3	10	10	30
10	1	15	18	7	20	1	8	22	10
	2	10	12	18		2	5	20	25
	3	20	15	15		3	14	16	20

ОГЛАВЛЕНИЕ

<i>ЛАБОРАТОРНАЯ РАБОТА №1</i> Оценка трудоемкости алгоритмов.....	3
<i>ЛАБОРАТОРНАЯ РАБОТА №2</i> Анализ характеристик функционирования ЦУС.....	10
<i>ЛАБОРАТОРНАЯ РАБОТА №3</i> Синтез систем оперативной обработки.....	25
<i>ЛАБОРАТОРНАЯ РАБОТА №4</i> 38 Расчет характеристик вычислительных сетей.....	38
<i>БИБЛИОГРАФИЧЕСКИЙ СПИСОК</i>	52
<i>ПРИЛОЖЕНИЯ</i>	53