

следите за обновлениями на

<http://fitu.narod.ru>

сканировал и правил Кондратов Сергей aka Serra
e-mail: serra@tut.by

СОДЕРЖАНИЕ

Лабораторная работа №1. Проектирование и анализ систем обслуживания.....	3
Лабораторная работа №2. Оценка и выбор вариантов сложных систем на этапе проектирования.....	15
Лабораторная работа №3. Моделирование организационной структуры управления предприятием.....	26
Лабораторная работа №4. Планирование вклада средств в производство.....	36
Лабораторная работа №5. Методы решения задач календарного планирования.....	42
Лабораторная работа №6. Оптимальное отображение множества функций системы по ее элементам.....	48
Лабораторная работа №7. Анализ и решение задач распределения ресурсов.....	56
Лабораторная работа №8. Построение, моделирование и анализ дискретных производственных систем.....	64
Литература.....	78

ЛАБОРАТОРНЫЙ ПРАКТИКУМ
по курсу " Основы автоматизированного управления "

для студентов специальности
"Автоматизированные системы обработки информации"

Никульшин Б.В., Иванчиков А.А., Сивиренцева О.Н. Лабораторный практикум по курсу " Основы автоматизированного управления "для студентов специальности "Автоматизированные системы обработки информации", Мн.: БГУИР, 1999.- 80с.

Практикум содержит описание лабораторных работ по курсу "Основы автоматизированного управления ", который читается студентам специальности "Автоматизированные системы обработки информации".

Предназначен для изучения прикладных методик и приобретения практических навыков проектирования автоматизированных систем обработки информации и управления моделями и методов принятия решений в автоматизированных системах обработки информации.

Лабораторные работы выполнены на базе ПЭВМ сотрудниками кафедры искусственных информационных технологий Белорусского государственного университета информатики и радиоэлектроники.

Ил. 37, табл. 10 , список литературы-23 назв.

ЛАБОРАТОРНАЯ РАБОТА № 1

ПРОЕКТИРОВАНИЕ И АНАЛИЗ СИСТЕМ ОБСЛУЖИВАНИЯ

1.1. Цель работы

1. Ознакомление с методами имитационного моделирования, применяемыми в процессе раннего проектирования систем обслуживания.
2. Программная реализация алгоритма модели систем обслуживания.

1.2. Общие сведения

1.2.1. Модель, моделирование

В настоящее время нельзя назвать область человеческой деятельности, в которой в той или иной степени не использовались бы методы моделирования. Особенно это относится к сфере проектирования, анализа и управления различными системами, где основными являются процессы принятия решений на основе получаемой информации.

Гипотезы и аналогии, отражающие реальный, объективно существующий мир, должны обладать наглядностью или сводиться к удобным для исследования логическим схемам: такие логические схемы, упрощающие рассуждения и логические построения или позволяющие проводить эксперименты, упрощающие природу явлений, называются *моделями*. Другими словами, модель (лат. *modulus* - мера) - это объект-заместитель объекта-оригинала, обеспечивающий изучение некоторых свойств оригинала.

Замещение одного объекта другим с целью получения информации о важнейших свойствах объекта-оригинала с помощью объекта-модели называется *моделированием*, Таким образом, моделирование может быть определено как представление объекта моделью для получения информации об этом объекте путем проведения экспериментов с его моделью.

Следует отметить, что с точки зрения философии моделирование - эффективное средство познания природы. Процесс моделирования предполагает наличие: объекта исследования; исследователя, перед которым поставлена конкретная задача; модели, создаваемой для получения информации об объекте.

Причем по отношению к модели исследователь является по сути дела экспериментатором, только в данном случае эксперимент проводится не с реальным объектом, а с его моделью. Такой эксперимент для инженера-системотехника - есть инструмент непосредственного решения организационно-технических задач.

Исторически первым сложился *аналитический подход* к исследованию систем, когда ЭВМ использовалась в качестве вычислителя по аналитическим зависимостям. Анализ характеристик процессов функционирования больших систем с помощью только аналитических методов исследования наталкивается обычно на значительные трудности, приводящие к необходимости существенного упрощения моделей либо на этапе их построения, либо в процессе работы с моделью, что может привести к получению недостоверных результатов.

Поэтому в настоящее время наряду с построением аналитических моделей большое внимание уделяется задачам оценки характеристик больших систем на основе *имитационных моделей*, реализованных на современных универсальных ЭВМ. При имитационном моделировании реализующий модель алгоритм воспроизводит процесс функционирования системы во времени, причем имитируются элементарные явления, составляющие процесс, с сохранением их логической структуры и последовательности протекания во времени, что позволяет по исходным данным получить сведения о состояниях процесса в определенные моменты времени, дающие возможность оценить характеристики системы. В настоящее время имитационное моделирование - наиболее эффективный метод исследования больших систем, а часто и единственный практически доступный метод получения информации о поведении системы, особенно на этапе ее проектирования.

Когда результаты, полученные при воспроизведении на имитационной модели процесса функционирования системы, являются реализациями случайных величин и функций, тогда для нахождения характеристик процесса требуется его многократное воспроизведение с последующей статистической обработкой информации.

1.2.2. Математическая схема систем массового обслуживания

Значительное место в теории имитационного моделирования занимают *системы массового обслуживания (СМО)*. СМО (*queuing system*) представляют собой класс математических схем, разработанных в теории массового

обслуживания и различных приложениях для формализации процессов функционирования систем, которые по своей сути являются процессами обслуживания [17].

В качестве процесса обслуживания могут быть представлены различные по своей природе процессы функционирования экономических, производственных, технических и других систем, например: потоки поставок продукции некоторому предприятию, потоки деталей и комплектующих изделий на сборочном конвейере цеха, заявки на обработку информации серверам от удаленных рабочих станций и т.д. При этом характерным для работы таких объектов является случайное появление заявок на обслуживание и завершение обслуживания в случайные моменты времени. Остановимся на основных понятиях теории массового обслуживания, необходимых для использования схем СМО как при аналитическом, так и При имитационном подходе.

В любом элементарном акте обслуживания можно выделить две основные составляющие: ожидание заявкой обслуживания и собственно обслуживание заявки. Это можно изобразить в виде некоторого прибора обслуживания Р (рис.1.1.), состоящего из накопителя заявок Н, в котором может одновременно находиться $l = 0, \dots, L$ заявок, где L - емкость накопителя, и канала обслуживания заявок (или просто канала) К. На каждый элемент прибора обслуживания Р

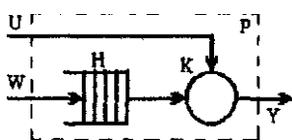


Рис. 1.1. Прибор обслуживания заявок

поступают потоки событий: в накопитель Н поток заявок w , на канал К - поток обслуживания i .

Потоком событий называется последовательность событий, происходящих одно за другим в какие-то случайные моменты времени.

Поток событий называется *однородным*, если он характеризуется только моментами наступления

этих событий (вызывающими моментами) и задается последовательностью $\{t_n\} = \{0 \leq t_1 \leq t_2 \leq t_n \leq \dots\}$, где t_n - момент наступления n -го события - неотрицательное вещественное число. Однородный поток событий также может быть задан в виде последовательности промежутков времени между n -м и $(n-1)$ -м событиями $\{\tau_n\}$, которая однозначно связана с последовательностью вызывающих моментов $\{t_n\}$, где $\tau_n = t_n - t_{n-1}$, $n \geq 1$, $t_0 = 0$, т.е. $t_1 = t_1$.

Потоком неоднородных событий называется последовательность $\{t_n, f_n\}$, где t_n - вызывающие моменты; f_n - набор признаков события. Например, применительно к процессу обслуживания для неоднородного потока заявок может

быть задана принадлежность к тому или иному источнику заявок, наличие приоритета, возможность обслуживания тем или иным типом канала и т.п. [17].

Обычно в приложениях при моделировании различных систем применительно к элементарному каналу обслуживания K можно считать, что поток заявок W (интервалы времени между моментами появления заявок на входе K) образует подмножество *неуправляемых переменных*, а поток обслуживания U (интервалы времени между началом и окончанием обслуживания заявки) образует подмножество *управляемых переменных*. Переменные называются управляемыми, если на их значение может воздействовать руководитель (лицо принимающее решение).

Заявки, обслуженные каналом K , и заявки, покинувшие прибор P и по различным причинам оставшиеся необслуженными (например, из-за переполнения накопителя H), образуют выходной поток Y .

Процесс функционирования прибора обслуживания P можно представить как процесс изменения состояний его элементов во времени $z(t)$. Переход в новое состояние для P означает изменение количества заявок, которые в нем находятся (в канале K и в накопителе H).

Таким образом, схема СМО, описывающая процесс функционирования системы любой сложности, однозначно задается в виде $СМО = \langle W, U, H, Z \rangle$.

Предметом исследований при моделировании СМО являются распределения системных величин, зависящих от потока заявок W и потока Обслуживания U . Некоторые из этих переменных перечислены ниже.

1. Число заявок, прибывших на обслуживание за заданный промежуток времени.
2. Число заявок, которые попали на обслуживание сразу же по прибытии.
3. Среднее время пребывания заявок в очереди.
4. Средняя длина очереди.
5. Максимальная длина очереди.
6. Нагрузка прибора (коэффициент его использования), являющаяся функцией времени, которое потрачено на обслуживание в течение заданного промежутка времени.

1.3. Простая модель СМО

Рассмотрим систему, состоящую из одного «прибора», выполняющего обслуживание определенного вида. Этот «прибор» может быть кассиром, продающим билеты на станции, парикмахером или же кладовщиком. «Клиенты» приходят к такому «обслуживаемому прибору» в случайные моменты времени, ждут своей очереди на обслуживание, их обслуживают по принципу «первый пришел - первым обслужен». После этого они уходят.

Используя процедурный подход, разработаем логическую схему модели на ЭВМ, которая будет имитировать систему обслуживания с одним прибором, двумя типами заявок и очередью. Схема должна удовлетворять следующим условиям.

1. Случайные переменные $IntA1$, $IntB1$, $IntA2$, $IntB2$, представляющие собой интервал прибытия заявок, являются равномерно распределенными и принимают только целые значения.

2. Времена обслуживания заявок прибором $ServA1$, $ServB1$, $ServA2$, $ServB2$ также являются равномерно распределенными случайными величинами, представлены целыми числами.

3. Есть возможность обращения к генератору случайных чисел, организованному в виде функции, возвращающей целые случайные числа в диапазоне $0 \dots n-1$.

4. Все прибывающие заявки должны быть обслужены независимо от длины очереди $Queue$.

5. В начале моделирования очередь пуста прибор свободен.

6. Моделирование продолжается до тех пор, пока не будет достигнуто значение модельного времени $Stop$, заданное для Модели в качестве одного из входных данных. Как только этот интервал времени истечет, моделирование будет завершено. В общем случае, когда моделирование завершается, обслуживающий прибор может быть в состоянии обслуживания заявки и в очереди может находиться одна или более заявок. Однако могут быть использованы и другие стратегии окончания моделирования. По одной из них прибор заканчивает работу не в момент времени $Stop$, а после обслуживания заявки. По второй стратегии прибор не начинает обслуживание очередной заявки, если планируемое время окончания ее обработки превысит значение $Stop$.

7. В процессе моделирования вся необходимая информация должна фиксироваться, а по окончании моделирования выдаваться на экран.

Работа модели напрямую связана с последовательным возникновением

событий и таймером модельного времени.

1.3.1. События

Все события, возникающие в системе, можно разделить на две категории, называемые *основными и вспомогательными событиями*. Основным называется такое событие, время возникновения которого можно запланировать заранее, т.е. рассчитать его до фактического возникновения. Вспомогательными событиями являются такие, время возникновения которых невозможно запланировать заранее.

В рассматриваемой системе основные события - это такие, как приход заявки и окончание обслуживания заявки; а вспомогательные - постановка заявки в очередь и поступление заявки на обслуживание.

1.3.2. Таймер

Когда начинается моделирование, таймер модельного времени Clock принимает нулевое значение, а затем, в процессе работы модели, его значение постоянно увеличивается. Предположим, что моделируется некоторая система, и состояние системы изменилось при текущем значении модельного времени. Следующим логическим шагом является увеличение значения таймера. Но на какую величину должно быть увеличено значение таймера? Существуют два метода решения этого вопроса.

Первый. Увеличить значение таймера на одну единицу времени. Затем проверить состояние системы и определить те из запланированных событий, которые должны произойти при новом значении таймера. Если таковые имеются, то выполнить их обработку, а затем вновь изменить значение таймера на одну единицу. Если же для нового значения таймера ни одного события ранее запланировано не было, то просто произойдет продвижение таймера на единицу времени. Такой метод основан на концепции фиксированного приращения значений таймера.

Во втором случае имеет место концепция переменного приращения значений таймера. При таком подходе условием, вызывающим приращение таймера, является наступление события, возникновение которого запланировано на ближайший к текущему значению таймера момент времени.

Бесспорное преимущество переменного приращения значений таймера объясняется тем, что при нем можно избежать обработки в промежуточные

моменты времени, когда не планируется выполнение никаких событий и, значит, не оказывается никакого влияния на состояние модели.

1.3.3. Датчик случайных чисел

При моделировании в связи со случайным характером изучаемых процессов возникает необходимость разыгрывать числа в соответствии с некоторым распределением. Для рассматриваемой модели необходимо иметь целые значения из выборок равномерно распределенных случайных чисел для задания интервалов прихода заявок и интервалов обслуживания. Удобным способом определения таких выборок является указание наименьшего и наибольшего целых чисел, содержащихся в выборке. Для определения такого рода выборки используется запись $[A..B]$. Часто используется и второй способ. Например, необходимо установить, что время обслуживания распределено равномерно и заключено в интервале $[18..30]$. Средним в этой выборке является 24, а размах составляет 6. Такой размах между средним и крайними значениями называют *половиной поля допуска*. Следовательно, можно для описания выборки указать среднее 24 и половину поля допуска 6. Недостатком такой записи, общий вид которой $[A \pm B]$, является невозможность задания четного числа значений. Поэтому предпочтение отдается первому способу определения случайной выборки равномерно распределенных чисел.

Если говорить о конкретной программной реализации, то источником случайных чисел может являться функция, возвращающая за каждый вызов очередное случайное число в заданном интервале. Такая функция называется генератором случайных (или, точнее, псевдослучайных) чисел. Чаще всего при проведении модельных экспериментов один и тот же генератор случайных чисел используется для определения различных параметров системы. Вследствие этого необходимо иметь промежуточную функцию, преобразующую интервал случайных чисел к нужному в данный момент. Ключевым выражением такой функции будет

$$A + \text{random}(B-A+1),$$

где функция $\text{random}(n)$ возвращает случайное число в интервале от 0 до $n-1$.

1.3.4. Алгоритмизация модели

На рис. 1.2. представлена блок-схема алгоритма решения задачи моделирования рассматриваемой СМО.

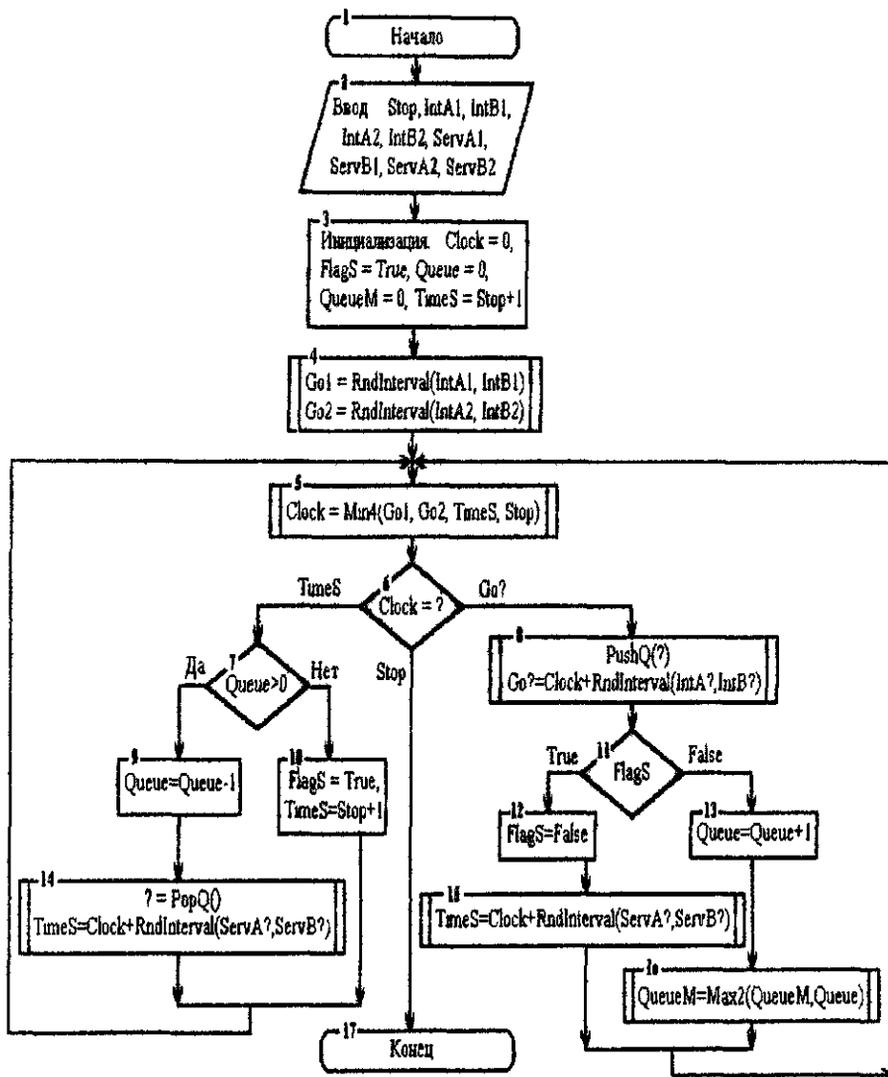


Рис.1 2. Блок-схема алгоритма обработки в системе обслуживания с одним прибором и очередью

В табл 1 1 перечислены и описаны все используемые в блок-схеме переменные и функции

Таблица 1 1

Переменная (функция)	Описание
Clock	Текущий таймер модельного времени
IntA1, IntA2	Значение А интервалов времени прихода заявок 1-го и 2-го типов
IntB1, IntB2	Значение В интервалов времени прихода заявок 1-го и 2-го типов
ServA1, ServA2	Значение А времени обслуживания заявок 1-го и 2-го типов
ServB1, ServB2	Значение В времени обслуживания заявок 1-го и 2-го типов
FlagS	Флаг состояния прибора (True - свободен, False - занят)
TimeS	Время конца обслуживания заявки прибором. Если прибор свободен, TimeS устанавливается в значение Stop+1
Stop	Время завершения моделирования
Queue	Длина очереди - число заявок, находящихся в очереди в текущий момент времени
QueueM	Максимальная длина очереди
Go1, Go2	Время прихода заявок 1-го и 2-го типов
RndInterval	Функция, принимающая параметры А и В и возвращающая случайное число в диапазоне А...В
Min4	Функция, принимающая четыре целых числа и возвращающая минимальное из них
Max2	Функция, принимающая два целых числа и возвращающая максимальное из них
PushQ	Функция, помещающая заявку в очередь (тип заявки передается параметром)
PopQ	Функция, возвращающая тип удаляемой из очереди заявки

Алгоритм функционирования модели достаточно прозрачен и может быть описан следующим образом

Блок 2 (см. рис. 1.2): читаются неуправляемые переменные модели.

Блок 3: инициализируются служебные переменные модели.

Блок 4: вычисляются времена прихода заявок 1-го и 2-го типов.

Блок 5: определяется время ближайшего события.

Блок 6: оценивается тип события, и процесс направляется по соответствующему руслу.

Блок 7: определяется наличие каких-либо заявок в очереди на обслуживание.

Блок 8: пришедшая заявка помещается в очередь и планируется время прихода следующей заявки данного типа.

Блок 9: из очереди удаляется одна заявка.

Блок 10: если очередь пуста, прибор освобождается.

Блок 11: определяется занятость прибора.

Блок 12: в случае если прибор свободен, он занимается.

Блок 13: если в момент прихода заявки прибор занят.

Блок 14: заявка помещается в очередь и планируется время конца обслуживания заявки.

Блок 15: планируется время конца обслуживания заявки.

Блок 16: определяется максимальная длина очереди.

Алгоритм реализует только базовые функции модели, и в нем отсутствуют блоки, позволяющие рассчитывать среднюю длину очереди, коэффициент использования прибора, блоки, управляющие приоритетом заявок, а также блоки определения некоторых других параметров модели (см. раздел 1.2.2).

1.3.5. Расширение существующего алгоритма

Среднее время пребывания заявок в очереди может быть рассчитано двумя способами. Первый способ учитывает заявки, которые поступают на обслуживание мгновенно (без помещения в очередь), второй - нет. В любом случае в модель вводится переменная `QueueSumm`, суммирующая все времена пребывания заявок в очереди. По окончании моделирования частное `QueueSumm/CountA` будет являться искомым средним временем (`CountA` - количество поступивших в очередь заявок). При помещении заявки в очередь время ее поступления запоминается в специальном массиве функцией, напоминающей `PushQ`. В момент, когда заявка покидает очередь и ставится на обслуживание, функция, похожая на `PopQ`, извлекает время поступления заявки

ClockOld и по разности Qlock-QlockOld определяется время нахождения заявки в очереди.

Для вычисления средней длины очереди необходимо через равные, относительно небольшие, промежутки времени оценивать длину очереди Queue, суммируя значение Queue в переменной QueueMedSumm. Для этого в программу вводится дополнительное событие (опрос очереди) с интервальными значениями времени наступления A...A. По окончании Моделирования частное QueueMedSumm/CountMed будет представлять собой значение средней длины очереди (переменная countMed - количество опросов очереди).

Коэффициент использования прибора вычисляется как частное TimeWork/Stop, где TimeWork - суммарное время работы станка, которое подсчитывается суммированием времен TimeS в момент постановки заявки на обслуживание.

Приоритет в обслуживании заявок учитывается сортировкой массива очереди при помещении туда заявки функцией PushQ.

1.4. Порядок выполнения работы

1. Изучить вопросы, связанные с имитационным моделированием СМО.
2. Досконально разобраться в алгоритме модели СМО и в соответствии с выбранным вариантом индивидуального задания описать используемые в алгоритме переменные (табл.1.1).
3. Разработать и на любом языке программирования написать программу моделирования выбранной СМО.
4. Расширить программу необходимыми вставками для сбора и вывода всей полной статистики (раздел 1.2.2).

Индивидуальные задания.

1. В парикмахерской могут находиться клиенты 2-х типов. Клиенты первого типа желают только стричься. Распределение интервалов их прихода 25...45 минут. Клиенты второго типа желают постричься и побриться. Распределение интервалов их прихода 40...80 минут. Парикмахер обслуживает клиентов в порядке их прихода. На стрижку уходит 12...24 минуты, а на бритье 8...12 минут. Написать модель работы парикмахерской в течение 8 часов для одного и двух мастеров.
2. На некоторой фабрике в кладовой работает один кладовщик. Он выдает

запасные части механикам, обслуживающим станки. Время, необходимое для удовлетворения запроса, зависит от запасной части. Запросы бывают двух категорий. За деталями первой категории механики приходят каждые 60...780 секунд и обслуживаются в течение 210...390 секунд. За деталями второй категории механики приходят каждые 120...600 секунд и обслуживаются в течение 70...130 секунд. Простой механика в очереди приносит убыток 0,25 цента в секунду. Определить, увеличится убыток фабрики или уменьшится, если; а) запросы второй категории будут иметь приоритет перед запросами первой категории; б) вместо первого кладовщика, получающего 4 доллара в час, будет работать другой, получающий 4,5 доллара в час и выполняющий заявки рабочих по первой категории заказов с интервалом 220...320 секунд, а второй категории - с интервалом 230.. .430 секунд.

1.5. Содержание отчета

1. Цель работы.
2. Условие решаемой задачи.
3. Определение переменных модели (табл. 1.1) в соответствии с выбранной задачей.
4. Характеристики рассматриваемой СМО по результатам проведенных модельных экспериментов (раздел 2.2).
5. Выводы по работе, содержащие конкретные результаты решаемых задач.

1.6. Контрольные вопросы

1. Что такое моделирование, и в каких областях человеческой деятельности оно применяется?
2. Что представляет собой СМО и каковы ее основные характеристики?
3. Смысл основных и вспомогательных событий в моделируемой системе.
4. Что представляет собой таймер модельного времени?
5. Какие стратегии розыгрыша случайных чисел могут применяться в имитационном моделировании?
6. Что представляет собой алгоритм моделирования процессов, протекающих в СМО?

ЛАБОРАТОРНАЯ РАБОТА №2

ОЦЕНКА И ВЫБОР ВАРИАНТОВ СЛОЖНЫХ СИСТЕМ НА ЭТАПЕ ПРОЕКТИРОВАНИЯ

2.1. Цель работы

1, Ознакомление с этапами выбора рациональных вариантов систем на ранних этапах их проектирования.

2. Многокритериальный выбор рационального варианта информационной подсистемы (ИП) АСУ воздушным движением (АСУ ВД) в диапазоне условий ее функционирования.

2.2. Общие сведения

ИП АСУ ВД предназначена для получения, обработки и выдачи потребителям информации о воздушных объектах (ВО), находящихся в зоне ответственности системы. Элементами ИП являются территориально разнесенные источники радиолокационной информации - радиолокационные станции (РЛС), замыкающиеся на пункты обработки информации (ПОИ) с помощью телекодовых трактов передачи данных. Для информационного обеспечения системы создается непрерывное радиолокационное поле (РЛП) путем рационального размещения станции в зоне ответственности, центрального управления ими и объединением информации от всех РЛС на ПОИ.

Достижение основной цели ИП - получение, обработка и выдача информации о ВО с требуемым качеством. Это осуществляется автоматизированным решением во всех звеньях ИП следующих задач: получение информации на РЛС о ВО и ее доведение до ПОИ; обработка и обобщение информации с помощью комплекса средств автоматизации (КСА) ПОИ от различных РЛС с целью получения более достоверных и качественных данных об ВО; передача полученных данных на вышестоящий КСА центрального пункта обработки информации для получения полной информационной картины о множестве ВО на всем временном интервале управления; управление системой с

целью рационального распределения ресурсов РЛС. Качество решения этих задач обуславливает качество функционирования ИП и выдаваемой информации [8].

ИП относится к системам с пространственно распределенной структурой, определение структуры которых заключается в выборе числа узлов (числа РЛС), их расположения, в определении взаимосвязи между узлами, распределении функций системы по узлам (распределение этапов обработки радиолокационной информации между РЛС и ПОИ), определении числа и типов РЛС, замыкающихся на ПОИ.

Выбор рационального структурного варианта базируется на анализе влияния его особенностей (структурных параметров) на качество функционирования системы в заданных условиях функционирования.

Выбор показателя качества функционирования основывается на анализе целевого назначения системы и согласовывается с заказчиком. Основным принципом формирования критерия выбора альтернатив является возможность количественной оценки проектных решений с учетом частных показателей качества функционирования, затрат и сроков создания системы [14]. Сроки создания системы могут быть переведены в разряд ограничений. Учет вектора показателей при сравнении вариантов наиболее удобен, иногда и единственно возможен, если вводится обобщенный показатель качества функционирования i -го варианта системы Q_i^u , на базе частных показателей $Q_i^u = Q(R_{11}^u, \dots, R_{ij}^u, \dots, R_{ij}^u)$, $i = 1, \dots, I$; $u = 1, \dots, U$. Здесь Q_i^u - оценка обобщенного показателя качества функционирования i -го варианта системы в условиях U ; $\{R_{11}^u, \dots, R_{ij}^u, \dots, R_{ij}^u\}$ - вектор показателей качества функционирования i -го ($i = 1, \dots, I$) варианта системы в условиях U , $u = 1, \dots, U$.

Существует довольно много методов построения обобщенного показателя.

В наиболее простых используются аддитивные $Q_i^u = \sum_{j=1}^J \alpha_j * R_{ij}^u$ и

мультипликативные $Q_i^u = \prod_{j=1}^J (R_{ij}^u)^{\alpha_j}$ преобразования над выбранной системой

частных показателей, где α_j , $j = 1 \dots J$ - коэффициенты, учитывающие веса показателей. Чаще всего при определении весов учитывается система предпочтений лица, принимающего решение (ЛПР), а также объективные факторы, отражающие относительный разброс оценок показателей по

факторы, отражающие относительный разброс оценок показателей по альтернативам [15].

Формируется критерий выбора альтернатив на основе обобщенного показателя и затрат. Под критерием выбора понимается признак (правило), по которому производится сравнительная оценка альтернатив и выбор наилучшей. В классической постановке "эффективность-стоимость" критерием выбора выступает какое-либо соотношение $F = F(Q_i^u, C_i)$, $u = 1, \dots, U$, где C_i - затраты на разработку 1-го варианта системы. В случае, когда варианты сопоставимы по важности с другими частными показателями или же малоразличимы по альтернативам, целесообразно включить C_i на правах частного показателя в обобщенный и формировать критерий выбора с учетом анализа условий функционирования системы. На этапе проектирования в лучшем случае располагают распределением вероятностей возможных условий, в которых предстоит функционировать системе, но чаще всего известен лишь их диапазон. В первом случае критерий выбора альтернатив оперирует со "взвешенными" оценками обобщенных показателей (с их математическими ожиданиями):

$$F = F(\bar{Q}_i) = \max_i \left(\min_j \right) \sum_{u=1}^U p(u) \cdot Q_i^u,$$

где $p(u)$ - вероятность того, что системе предстоит функционировать в условиях u . Во втором случае привлекаются специальные критерии из теории игр - Вальда, Лапласа, Гурвица, Сэвиджа и другие [16]. При этом выбор рациональной структуры осуществляется на основе анализа игровой матрицы, в которой стратегиями являются варианты структуры и условия функционирования, а мерой платежа - обобщенные показатели качества функционирования (табл.2.1):

Таблица 2.1

	Варианты условий				
V_i	1	...	u	...	U
V_i	Q_i^1	...	Q_i^u	...	Q_i^U
...
V_i	Q_i^1	...	Q_i^u	...	Q_i^U
...
V_i	Q_i^1	...	Q_i^u	...	Q_i^U

2.3 Постановка задачи выбора рациональной структуры ИП на ранних стадиях проектирования.

Задача структурной оптимизации ИП АСУ ВД на ранних стадиях проектирования рассматривается в плане выбора рационального варианта расположения станций и ПОИ в зоне ответственности ИП (задача выбора типологии ИП) с учетом качества трактов передачи.

Анализ альтернативных вариантов ИП осуществляется в диапазоне условий функционирования системы. Варианты условий отличаются длительностями и ориентацией в пространстве относительно станций обслуживаемых маршрутов ВО в зоне ответственности ИП, а также погодными условиями функционирования системы. В терминах служб АСУ ВД эти условия интерпретируются требуемым временем слежения за ВО системой T_u , $u=1, \dots, U$ и вероятностью правильного обнаружения ВО каждой из РЛС ИП $P_{\text{об}}^m$, $m=1, \dots, M$, где M - число станций системы.

Место расположения РЛС определяет РЛП ИП. Под РЛП понимается пространство с определенными высотами нижней и верхней границ, где ВО обнаруживается станциями с заданной вероятностью. Многослойность РЛП определяется коэффициентом перекрытия K_p , численно равным числу РЛС "видящих" в данной зоне пространства ВО с заданной вероятностью. В свою очередь РЛП наряду с другими факторами оказывает существенное влияние на одну из важнейших характеристик функционирования ИП - вероятность слежения за ВО системой станций в течение времени T $W(T)$.

Анализ вариантов ИП проводится с помощью обобщенных показателей качества функционирования ИП

$$Q_i^u(K_p, W_i^u(T_u), C_i), \quad i = 1, \dots, I, \quad u = 1, \dots, U,$$

сформированных на основе аддитивных преобразований частных показателей. Веса частных показателей определяются на базе системы предпочтений ЛПР.

Выбор рационального варианта ИП базируется на анализе игровой матрицы с помощью критериев Вальда и Лапласа.

2.4 .Методика оценки эффективности ИП АСУ ВД

В качестве средств получения информации о находящихся в воздухе самолетах в различных контурах и комплексах используется система радиолокационных станций. Алгоритм действий, осуществляемый с помощью этой системы, следующий: по данным РЛС следует сопровождать каждый находящийся в воздухе самолет в течение времени, необходимого для их целераспределения, наведения или сопровождения. Это время называется временем исполнения (T), под исполнением понимается слежение за самолетом в течение промежутка, равного или большего T . Временной интервал, при котором происходит срыв процесса слежения, называется интервалом дискретности и обозначается dT . Задачей является определение вероятности проводки самолета $W_{пр}$ (вероятность слежения в течение времени T или большего) в зависимости от параметров системы РЛС и величин T и dT .

Систему РЛС, формирующих радиолокационное поле, можно охарактеризовать суммарном числом обнаружений λ одного самолета этим полем в единицу времени:

$$\lambda = \sum_{m=1}^M v^{(m)} P_0^{(m)},$$

где M - количество РЛС, формирующих поле; $v^{(m)}$ - число локаций самолета m -й РЛС в единицу времени; $P_0^{(m)}$ - вероятность обнаружения одной локации m -й РЛС.

Для случая, когда для сопровождения ВО достаточно на обзоре получить хотя бы одну одну отметку из возможного их числа M , для вычисления λ используется формула

$$\lambda = \sum_{m=1}^M v^{(m)} \left(1 - \prod_{k=1}^M \left(1 - P_0^{(m)} \cdot P_k^{(m)} \right) \right),$$

где $P_k^{(m)}$ - вероятность неискаженного доведения информации от m -й РЛС к ПОИ по телекодovому тракту передачи данных.

Рассматриваем здесь такие системы, в которых на приемных устройствах РЛС помимо детектирования осуществляется первичная обработка данных информации, а на центральный пункт обработки информация передается после принятия решения о наличии сигнала на каждой временной позиции и

обработке сигналов, а λ в этом случае является плотностью потока обнаружения единичной цели на входе уефойства вторичной обработки информации. На основании предельных теорем для редующих и суммарных потоков можно утверждать, что поток, плотность которого λ , близок к простейшему, го есть интервал между двумя соседними обнаружениями распределен по экспоненциальному закону $f(\tau) = \lambda \cdot e^{-\lambda \tau}$

Вероятность проводки одного самолета определяется методами теории массового обслуживания. Всю совокупность РЛС, информация от которых объединяется для сопровождения самолета, можно рассматривать как канал обслуживания, на вход которого поступает поток заявок с интенсивностью K Каждая заявка - одно обнаружение. Если заявки следуют с интервалами, меньшими dT , то сопровождение считается непрерывным Если интервал между заявками больше или равен dT , то происходит срыв сопровождения. Вероятность слежения системой за ВО в течение времени T_u определяется по формуле

$$W_i^u(T_u) = \exp\left(-\lambda T_u \cdot e^{-\lambda dT}\right).$$

При проектировании не располагают в полной мере возможным числом сопровождаемых ВО с их возможными маршрутами, проходящими по зонам с различными коэффициентами K_p (рис.2.1). В связи с этим целесообразно при

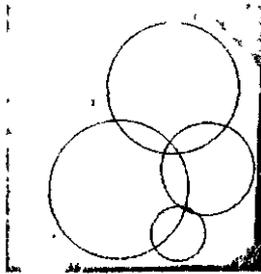


Рис.2.1

сравнительном анализе вариантов оперировать с усредненными по зонам РЛП с конкретными K_p оценками вероятности слежения системой за ВО, рассчитываемыми для предельных значений $T, u = 1 \dots U$:

$$\bar{W}_i^u(T_u) = \sum_{v=1}^V W_{iv}^u(T_u) \cdot \frac{S(v)}{S},$$

где V - множество РЛП с различными коэффициентами перекрытия; $S(v)$ - площадь v -зоны с $K_p = K_{pi}(v)$; S - общая площадь РЛП в зоне ответственности ИП, включая и "не просматриваемые" зоны $K_p = 0$.

ИП, включая и "не просматриваемые" зоны $K_p = 0$.

Коэффициент перекрытия РЛП в зоне ответственности ИП может быть определен по формуле:

$$\bar{K}_{Pi} = \sum_{v=1}^V K_{Pi}(v) \cdot \frac{S(v)}{S}$$

Оценка площадей получаемых "зон" производится методом статистических испытаний. При этом $S(v) = \frac{n(v)}{n}$, где n -общее число сгенерированных точек с помощью датчика случайных чисел, $n(v)$ - число точек, попавших в v -зону

Очевидно, что при такой методике оценки площадей зон важное значение для более качественной оценки имеет количество статистических испытаний. Чем больше их будет проведено, тем качественней оценка будет получена.

Стоимость разработки i -го варианта рассчитывается по формуле

$$C_i = C_{рлс} \cdot M + C_{пои} + C_k \cdot L_{ki},$$

где $C_{рлс}$ и $C_{пои}$ - соответственно стоимость одной РЛС и ПОИ; C_k - стоимость единицы длины телекодového тракта Передачи информации; L_{ki} - длина тракта передачи данных в единицах длины тракта для i -го варианта системы

Сравнительный анализ вариантов ИП проводится с помощью обобщенных показателей качества функционирования ИП, сформированных на основе аддитивных преобразований частных показателей. Веса частных показателей определяются на базе неформальных предпочтений ЛПР.

Для определения весов частных показателей используем одну из возможных модификаций метода парных сравнений:

1) составляется матрица бинарных предпочтений (табл.2.2), в которой

Таблица 2.2

Матрица бинарных предпочтений

R_j/R_k	R_1	R_2	R_k	...	R_j	f_j
R_1	###	0	0	...	0,5	f_1
R_2	1	###	1	...	1	f_2
R_k	1	0	###	...	0	f_j
	###
R_j	0,5	0	1	...	###	f_j

предпочтительнее, чем R_j ; 0 - в обратном случае; 0.5 - если оба показателя равнозначны;

2) определяется "цена" показателя f_j , $j = 1, \dots, J$ путем суммирования переменных в строке;

3) исходя из "цен" показателей вычисляются искомые веса частных показателей:

$$a_j = \frac{f_j}{\sum_{j=1}^J f_j}.$$

Обобщенный показатель качества функционирования ИП формируется по алгоритму:

1) формируется матрица "показатели - структуры", представленная в габл.2.3;

Таблица 2.3

Матрица "показатели-структуры"

R_j	Варианты структур				
	I	...	i	...	I
R_1	R_{1I}	...	R_{1i}	...	R_{1I}
...
R_j	R_{jI}	...	R_{ji}	...	R_{jI}
...
R_J	R_{jI}	...	R_{jI}	...	R_{jI}

2) оценки матрицы "показатели - структуры" приводятся к безразмерному виду:

$$r_{ij}^H = \frac{R_{ij}^H}{\max_i R_{ij}^H}, j \in J',$$

$$r_{ij}^H = \frac{R_{ij}^H}{\min_j R_{ij}^H}, j \in J'';$$

где J' - множество показателей, подлежащих максимизации, J'' - подлежащих минимизации;

3) формируется матрица мер качества функционирования

3) формируется матрица мер качества функционирования

$$\bar{r}_{ij}^u = a_j r_{ij}, i=1, \dots, I; j=1, \dots, J;$$

4) вычисляются обобщенные показатели ИП:

$$Q_i^u = \sum_{j \in J'} \bar{r}_{ij}^u - \sum_{j \in J''} \bar{r}_{ij}^u,$$

т.е. находится разность суммарных мер качества функционирования по показателям, подлежащим соответственно максимизации и минимизации.

Выбор рационального варианта ИП базируется на анализе игровой матрицы с помощью критериев Вальда и Лапласа, в которой стратегиями являются варианты структуры и условий функционирования, а мерой платежа - обобщенные показатели качества функционирования [2].

Критерий Вальда, или критерий осторожного наблюдателя,

$$F = \max_i \min_u Q_i^u$$

гарантирует определенный выигрыш при наихудших условиях. Оптимальная структура определяется следующим образом. Построчно по табл.2.4 отыскиваются минимальные оценки $\min_u Q_i^u, i=1, \dots, I$ для всех структур. Предпочтение имеет альтернатива с максимальным строчным минимумом.

Критерий Лапласа (предполагается, что все варианты условий равновероятны):

$$F = \max_i \frac{1}{U} \sum_{u=1}^U Q_i^u$$

Предпочтительнее структура с максимальной средней построчной оценкой.

2.5. Порядок выполнения работы

Программное обеспечение, используемое при выполнении работы, имеет развитое меню, сопровождающее пользователя в процессе проработки альтернатив в режиме активного диалога. Это освобождает пользователя от рутинных промежуточных расчетов по анализу вариантов информационной

определения условий функционирования системы и исходных данных для расчетов, формирования системы предпочтений ЛПР, управления исследованием и анализом их результатов.

Весь процесс исследования альтернатив отражается на экране монитора с фиксацией формирующего кадра до конца работы. На экране вся информация располагается в соответствующих окнах: окно для графической иллюстрации топологии размещения вариантов информационной подсистемы; окно для условий функционирования; контекстуально-зависимое окно подсказки; рабочее окно для проведения расчетов и вывода результатов сравнительного анализа альтернатив.

В соответствии с разработанным программным обеспечением в процессе исследования вариантов информационной подсистемы приняты следующие ограничения:

1) число радиолокационных станций каждого варианта может меняться от одного до пяти;

2) одновременно можно анализировать не более четырех вариантов ИП при трех вариантах условий функционирования системы.

Для оценки площадей зон радиолокационного поля используется метод статистических испытаний - метод Монте-Карло.

Лабораторная работа выполняется в следующей последовательности.

1. Изучить общие положения проектирования ИП АС УВД и составить набор альтернатив ИП, варьируя числом и расположением РЛС и ПОИ в зоне ответственности.

2. Сформировать несколько вариантов ИП АС УВД, включая формирования зоны ответственности (стандартная или определяемая пользователем) и расстановку в зоне ответственности станции,

3. Сформировать внешние условия функционирования ИП АС УВД: вероятность обнаружения ВО m -РЛС в условиях u - P_{ou}^m , $u = 1,2,3$, $m = 1,\dots,5$; T_u , $u = 1,\dots,3$; dT_u , $u = 1,\dots,3$.

4. Рассчитать характеристики ИП: коэффициент перекрытия; вероятность непрерывного сопровождения; стоимость.

5. Провести сравнительный анализ, вариантов ИП АС УВД и выбор наиболее предпочтительного.

наиболее предпочтительного.

2.6. Составление отчета

1. Цель работы.
2. Список и характеристики альтернатив, условий их функционирования.
3. Этапы анализа альтернатив и результаты расчетов по этапам.
4. Игровая матрица для определения рациональной структуры ИП и результаты анализа.
5. Выводы по работе.

2.7. Контрольные вопросы

1. Чем объясняется необходимость оптимизации процессов проектирования сложных систем на ранних этапах их разработки.
2. Определите целесообразность ввода обобщенных показателей качества функционирования сложных систем на этапе анализа проектных решений.
3. Приведите основные методы формирования обобщенных показателей качества функционирования систем. Дайте их сравнительный анализ.
4. Основные этапы выбора рациональных вариантов систем на ранних этапах проектирования в диапазоне возможных условий их функционирования.
5. В чем смысл критериев выбора альтернатив Вальда, Сэвиджа, Гурвица, Лапласа?

Лабораторная работа №3

Моделирование организационной структуры управления предприятием

3.1. Цель работы

1. Ознакомление с методиками формирования организационных структур управления предприятием.
2. Определение оптимального количества уровней линейного и функционального управления организационной структуры.
3. Определение оптимального разбиения структурного подразделения на отдельные группы.

3.2. Общие сведения

Организационная структура управления - это состав, взаимосвязи и соподчиненность совокупности организационных единиц (подразделений) аппарата управления, выполняющих различные функции по управлению отраслью, производственным объединением (предприятием), цехом.

Важнейшими задачами при формировании организационной структуры являются, во-первых, определение оптимального количества уровней управления, и, во-вторых, оптимального разбиения структурных подразделений на самостоятельные группы на каждом из уровней.

Организационная структура аппарата управления предприятием имеет пирамидальный характер, т.е. содержит несколько уровней управления (иерархических), что соответствует принципам системного построения организации. Иерархичность организационной структуры обеспечивает ее устойчивость к возмущающим воздействиям, способствует ликвидации возможных конфликтных ситуаций путем соблюдения строгой соподчиненности работников и руководителей.

Научное обоснование рационального количества уровней управления и распределения по ним управленческих работников - важный фактор, от которого зависит эффективность работы аппарата управления. Излишнее количество уровней управления замедляет движение управленческой информации, в то же время короткие иерархические цепочки или недостаточная численность работников на каком-либо уровне управления приводят к информационной перегрузке работников на вышестоящих уровнях.

Иерархическое построение аппарата управления тесно связано с централизацией управления - сосредоточением организационных и распорядительных функций на высших уровнях управления и исполнительских - на нижних уровнях. Преимущества централизации: четкое распределение обязанностей в организации, исключение параллелизма в работе; его главный

недостаток - снижение гибкости и оперативности работы аппарата управления. Децентрализованное управление позволяет находить оптимальные решения на всех уровнях управления, уменьшает объем информации, перерабатываемой на верхних уровнях; его недостаток - затруднение координации, регулирования и контроля процессов управления.

Применение экономико-математической модели иерархии организационной структуры позволяет оптимизировать один из важнейших параметров организационной структуры - степень централизации управления - путем определения оптимального количества уровней линейной (например, директор - начальник цеха - мастер) и функциональной (например, главный инженер - главный конструктор - начальник бюро - руководитель группы) управления и распределения работников управления между ними.

Руководитель, осуществляя распорядительно-контрольную функцию, отдает распоряжения подчиненным и контролирует их выполнение. Распорядительно-контрольная функция количественно выражается числом распоряжений (указаний, инструкций) за определенный промежуток времени и затратами времени на выполнение распоряжения и его контроль. Такие данные могут быть получены при проведении специального исследования или целевой фотографией рабочего времени руководителей и исполнителей.

Если исследование проводится в форме анкетирования руководителей и исполнителей, то руководители в своих анкетах указывают Подчиненных и распорядительно-контрольную функцию для каждого из них в абсолютных и относительных величинах. Подчиненные в анкетах указывают работников вышестоящего уровня, от которых они получают распоряжения, и величину этой связи в абсолютных или относительных показателях для всех названных руководителей.

Сведения по каждой серии таких анкет могут быть представлены в форме

Руководители

		<i>Руководители</i>											
		<i>t = 1</i>			<i>t = 2</i>			<i>t = 3</i>					
<i>Исполнители</i>		<i>i</i>	<i>11</i>	<i>21</i>	<i>22</i>	<i>23</i>	<i>31</i>	<i>32</i>	<i>33</i>	<i>34</i>	<i>35</i>	<i>36</i>	
		<i>n = 2</i>	<i>i</i>	<i>11</i>	0	0	0	0	0	0	0	0	0
<i>21</i>	0,2		0	0	0	0	0	0	0	0,3	0	0	
<i>22</i>	0,4		0,2	0	0	0	0	0,1	0	0	0	0	
<i>23</i>	0,4		0,2	0	0	0	0	0,1	0	0	0	0	
<i>n = 1</i>	<i>i</i>	<i>11</i>	0,2	0	0,7	0	0	0	0	0	0	0	
	<i>21</i>	0	0	0,1	0,8	0	0	0	0	0	0	0	
	<i>22</i>	0,1	0	0,1	0	0,8	0	0	0	0	0	0	
	<i>23</i>	0	0,8	0	0,1	0	0	0	0,1	0	0	0	

Рис.3.1.

прямоугольных матриц A_{tn} и A'_{tn} (рис.3.1), где t и n - соответственно число руководителей и подчиненных, а каждый элемент a_{ij} определяет связь по распорядительно-контрольной функции между i -м работником, указанным в крайней верхней строке матрицы, и j -м работником, указанным в крайнем левом столбце. Матрица A_{tn} составляется руководителями, A'_{tn} - подчиненными. Для усреднения полученных величин и учета взаимной оценки руководителей и исполнителей необходимо объединить обе матрицы A_{tn} и A'_{tn} в одну B_{tn} , перемножив между собой одноименные элементы матриц ($a_{ij} \cdot a'_{ij} = b_{ij}$). Такая процедура усиливает основные связи между работниками различных рангов и ослабляет случайные связи или ошибочные сведения, так как если один из сомножителей равен нулю, то и их произведение равно нулю, и связь остается не подтвержденной.

Построение графа фактической иерархической структуры аппарата управления U на основе объединенной матрицы B_{tn} производится путем распределения работников по подразделениям, а подразделений - по уровням управления, соответствующих рангу их руководителей. Исходный граф U соподчиненных связей в аппарате управления является, как правило, связным графом с некоторым количеством циклов. Чтобы получить из него искомый граф типа "дерево" G , отображающий оптимальную иерархическую структуру управления, необходимо из графа U удалить не менее v ребер. Величина v , называемая цикломатическим числом графа, определяется из соотношения

$$V = N - n + 1,$$

где N - количество ребер графа (число соподчиненных связей); n - число вершин графа (количество работников в аппарате управления).

Рациональной иерархии аппарата управления ставится в соответствие Такое дерево G , которое обладает общей максимальной суммой весов ребер и имеет наименьшую длину цепи от корня дерева до наиболее удаленной висячей вершины. Под весом ребра понимается количественная характеристика распорядительно-контрольной функции между двумя работниками (которые условно представляются в виде вершин графа), принадлежащими данному ребру графа.

Варианты иерархической структуры сравниваются по критерию оптимальности Q , который учитывает сумму весов всех ребер графа b_i и максимальную длину цепи от корня дерева G до его наиболее удаленной висячей вершины:

$$Q = \frac{1}{R} \sum_{i=1}^N b_i \rightarrow \max ,$$

где R - максимальная длина цепи от корня дерева G до его наиболее удаленной висячей вершины, выраженная количеством ребер; b_i - вес i -го ребра графа G , т.е. показатель распорядительно-контрольной функции между руководителем и исполнителем.

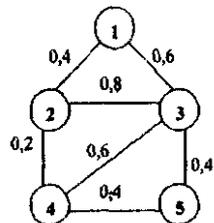


Рис.3.2. Исходный граф U

Тот вариант графа G, который обеспечивает максимальное значение критерия Q_{\max} , является искомым, откуда для данных условий определяется величина R, показывающая рациональное число уровней иерархии в аппарате управления.

Рассмотрим применение модели на примере рационализации иерархической структуры службы технологической подготовки производства на машиностроительном предприятии. В составе подразделений имеется бюро планирования производства в составе 5 человек. Результаты обследования соподчиненных связей по распорядительно-контрольной функции между этими работниками представлены в виде исходного графа U (рис.3.2). Его цикломатическое число

$$v=7-5+1=3,$$

т.е. для получения искомого графа G следует удалить не менее трех ребер у графа U.

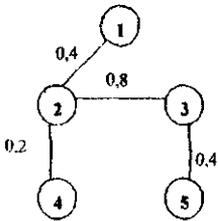


Рис 3.3. $R=3; \sum b_i=1,8; Q=0,6$

Первый вариант перестройки исходного графа U в дерево O (рис.3.3) характеризуется следующими показателями: максимальная длина от корня дерева G до его наиболее удаленной вершины, выраженная числом ребер, $R_1=3$; суммарная количественная характеристика распределительно-контрольной функции подсчитывается как сумма весов всех ребер

графа G и составляет $b_1=0,4+0,8+0,2+0,4=1,8$; критерий эффективности для данного варианта построения иерархической структуры $Q_1=1,8/3=0,6$.

Аналогичным образом производится оценка всех возможных вариантов построения иерархической структуры для данных исходных условий (рис.3.4-3.8).

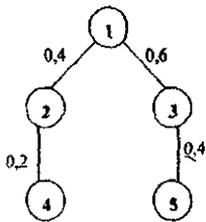


рис 3.4 $R=2; \sum b_i=1,6; Q=0,8$

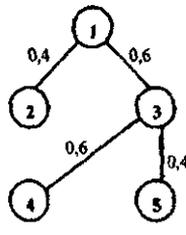


Рис.3.5. Искомый граф
Gopt $R=2; \sum b_i=2,0; Q=1,0$

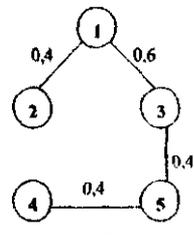


Рис.3.6. $R=3; \sum b_i=1,8; Q=0,6$

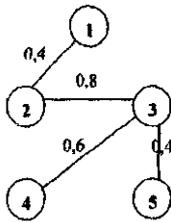


Рис.3.7 $R=3; \sum b_i=2,2; Q=0,73$

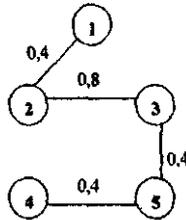


Рис.3.8. $R=4; \sum b_i=2,0; Q=0,5$

Как видно из рисунков, искомым графом Горт, соответствующим рациональному варианту построения иерархической структуры, является граф, показанный на рис.3.5, для которого установленный критерий эффективности имеет максимальное значение $Q=1,0$.

На основе проведенных расчетов по модели иерархической структуры представляется целесообразным для бюро планирования производства выделить отдельной группы (работники 3, 4, 5) с установлением для них самостоятельного круга вопросов.

Вторая задача формирования организационных структур состоит в формировании обособленных групп работников управления с учетом их кооперируемое при решении конкретных задач управления. При этом необходимо стремиться к формализации процесса структуризации и количественному сравнительному анализу вариантов разбиения. Ниже рассмотрим один из методов решения этой задачи [1].

Моделирование формирования самостоятельных структурных подразделений аппарата управления основывается на формализации производственных взаимосвязей между работниками управления, профессионально-квалификационного состава работников, характера и степени их загруженности выполнением работ по данной функции управления.

Исходное множество $\{M_0\}$ элементов $m \in \{M_0\}$, отображающее состав работников аппарат управления, разбивается на некоторое количество пустых подмножеств - самостоятельных структурных подразделений $\{M_i\} \subseteq \{M_0\}$, причем $\{M_i\} \neq \emptyset$. Сумма и пересечение этих подмножеств p а $\sum_{i=1}^n M_i = \{M_0\}$ и $\{M_{i1}\} \cap \{M_{i2}\} = \emptyset$ при $i1=i2$ (ни один работник не входит одновременно в два подразделения).

Элементы исходного множества $\{M_0\}$ находятся во взаимосвязи. В качестве показателей связи выступают количественные характеристики производственных взаимосвязей между работниками в процессе их совместного труда. К таким взаимосвязям относятся: взаимный обмен информацией, консультации, справки, отчеты, совещания. Большинство этих взаимосвязей может быть оценено количественно: удельным весом в балансе рабочего времени, трудоемкостью или частотой повторения за определенный промежуток времени.

Рациональная структуризация аппарата управления должна быть направлена на увеличение и укрепление кооперированных взаимосвязей. В то же время, развивая и углубляя кооперацию труда в аппарате управления и обеспечивая условия для этого в виде функционального обособления кооперируемых работников в самостоятельных структурных подразделениях, следует иметь в виду, что этот процесс приводит к дроблению аппарата управления на все более мелкие структурные единицы, что неизбежно порождает трудности в его функционировании. Следовательно, необходимо стремиться к

трудности в его функционировании. Следовательно, необходимо стремиться к тому, чтобы структуризация аппарата управления осуществлялась в минимально необходимых пределах, обеспечивая эффективное выполнение заданных функций управления.

Целевая функция Δ , соответствующая этим условиям, будет иметь вид

$$\Delta = \sum_{i=1}^n \underbrace{\sum_{j=1}^n m_{ij}}_{\beta} \underbrace{[\ln(t_i + 1)]}_{\gamma} \rightarrow \max,$$

где n - количество элементов m_i в подмножестве $\{M_i\}$, т.е. численность работников формируемого подразделения; t - количество производных подмножеств $\{M_i\}$; t_i - величина зафиксированной связи между элементами i и j ; m_{ij} - число самостоятельных структурных подразделений в аппарате управления, равное количеству производных подмножеств $\{M_i\}$. Выражение $(n+1)$ выбрано исходя из свойств логарифмической функции, у которой единственным нулем является точка $l=1$.

Показатели взаимосвязи t_{ij} между элементами исходного множества $\{M_0\}$ образуют квадратную матрицу $\|m_{ij}\|$, где каждый элемент характеризует величину производственной взаимосвязи между i -м элементом верхней строки и j -м элементом левого столбца матрицы $\|m_{ij}\|$.

Определение оптимальной структуризации аппарата управления на основе моделирования разделения и кооперации управленческого труда осуществляется путем перестройки исходной матрицы взаимосвязей $\|m_{ij}\|$ и оценки суммарной величины кооперированных связей β и сложности γ образуемой при этом организационной структуры. Рассмотрим практическую реализацию данной модели на примере фрагмента организационной структуры аппарата управления, где в качестве исходного множества $\{M_0\}$ взяты четыре управленческих работника. Установленные величины производственных взаимосвязей между ними отражены в исходной матрице (рис.3.9). Для удобства расчетов и наглядности величины кооперированных связей (среднегодовая трудоемкость в нормо-часах) несколько округлены и взяты с масштабным коэффициентом $1/100$.

Суммарный показатель кооперированных связей P для исходной матрицы $M_0 = \|m_{ij}\|$ составляет

$$\beta_0 = 5+2+3+1+1+2+4+1+2=24.$$

Показатель сложности исходной организационной структуры γ определяется из соображения, что в данном случае количество производных подмножеств $t=1$, следовательно,

$$\gamma_0 = \ln(1+1) = \ln 2 = 0,693.$$

$$\Delta_0 = 24 \cdot 0,693 = 16,632.$$

Исходная матрица M_0

работ ники	1-й	2-й	3-й	4-й	
1-й	X	5	2	3	$\beta=24$ $\gamma=0,693$ $\Delta=16,632$
2-й	3	X	0	1	
3-й	1	0	X	2	
4-й	4	1	2	X	

4-й вариант (1)(2,3,4)

работ ники	1-й	2-й	3-й	4-й	
1-й	X	-	-	-	$\beta=6$ $\gamma=1,091$ $\Delta=6,546$
2-й	-	X	0	1	
3-й	-	0	X	2	
4-й	-	1	2	X	

1-й вариант (1,2)(3,4)

работ ники	1-й	2-й	3-й	4-й	
1-й	X	5	-	-	$\beta=12$ $\gamma=1,091$ $\Delta=13,092$
2-й	3	X	-	-	
3-й	-	-	X	2	
4-й	-	-	2	X	

5-й вариант (2)(1,3,4)

работ ники	1-й	2-й	3-й	4-й	
1-й	X	-	2	3	$\beta=14$ $\gamma=1,091$ $\Delta=15,274$
2-й	-	X	-	-	
3-й	1	-	X	2	
4-й	4	-	2	X	

2-й вариант (1,3)(2,4)

работ ники	1-й	2-й	3-й	4-й	
1-й	X	-	2	-	$\beta=5$ $\gamma=1,091$ $\Delta=5,455$
2-й	-	X	-	1	
3-й	1	-	X	-	
4-й	-	1	-	X	

6-й вариант (3)(1,2,4)

работ ники	1-й	2-й	3-й	4-й	
1-й	X	5	-	3	$\beta=17$ $\gamma=1,091$ $\Delta=18,547$
2-й	3	X	-	1	
3-й	-	-	X	-	
4-й	4	1	-	X	

3-й вариант (1,4)(2,3)

работ ники	1-й	2-й	3-й	4-й	
1-й	X	-	-	3	$\beta=7$ $\gamma=1,091$ $\Delta=7,637$
2-й	-	X	0	1	
3-й	-	0	X	-	
4-й	4	-	-	X	

7-й вариант (4)(1,2,3)

работ ники	1-й	2-й	3-й	4-й	
1-й	X	5	2	-	$\beta=11$ $\gamma=1,091$ $\Delta=12,001$
2-й	3	X	0	-	
3-й	1	0	X	-	
4-й	-	-	-	X	

Рис.3.9. Пример реализации модели выделения самостоятельных структурных подразделений

Первый вариант структуризации исходного множества $\{M_0\}$ и выделения в нем самостоятельных подразделений показан на рис.3.9. Перестроение исходной матрицы взаимосвязей $M_0 = \|m_{ij}\|$ дает следующее: первое производное подмножество $\{M_1\}$ и образуется путем объединения первую и второго работников, а второе $\{M_2\}$ - третьего и четвертого. При этом исключаются из рассмотрения кооперированные взаимосвязи между элементами подмножеств $\{M_1\}$ и $\{M_2\}$

а второе $\{M_2\}$ - третьего и четвертого. При этом исключаются из рассмотрения кооперированные взаимосвязи между элементами подмножеств $\{M_1\}$ и $\{M_2\}$

Величина показателя β из образованной матрицы взаимосвязей $M_1 = \|m_{ij}\|$

$$\beta_1 = 5 + 3 + 2 + 2 = 12.$$

Показатель сложности образуемой организационной структуры γ при $t=2$ составит

$$\gamma_1 = \ln(2+1) = \ln 3 = 1,091.$$

Критерий эффективности Δ такого варианта построения организационной структуры

$$\Delta_1 = 12 \cdot 1,091 = 13,092.$$

Сопоставление величин $\Delta_1 < \Delta_0$ свидетельствует о нецелесообразности формирования двух самостоятельных подразделений при рассмотренном составе работников.

Следующая матрица $M_2 = \|m_{ij}\|$ получена из исходной $M_0 = \|m_{ij}\|$ путем образования двух подмножеств: $\{M_1\}$ - объединением первого и третьего работников и $\{M_2\}$ - второго и четвертого работников и исключением из расчетов взаимосвязей между выделенными подразделениями.

Показатель β второго варианта структуризации исходного множества $\beta_2 = 2 + 1 + 1 + 1 = 5$.

Показатель сложности образуемых подмножеств при $t=2$ составит

$$\gamma_2 = \ln(2+1) = \ln 3 = 1,091.$$

Критерий эффективности Δ для второго варианта $\Delta_2 = 5 \cdot 1,091 = 5,455$, т.е. $\Delta_2 < \Delta_1 < \Delta_0$.

Аналогичным образом производятся расчеты по всем возможным вариантам перестроения исходной матрицы взаимосвязей $M_0 = \|m_{ij}\|$. Как видно из рисунка, максимальным значением критерия эффективности характеризуется шестой вариант. Поэтому представляется целесообразным выделение самостоятельного структурного подразделения в рассматриваемой службе в составе трех работников (первого, второго и четвертого), а третий работник по функциональной принадлежности тяготеет, вероятно, к другой группе, не вошедшей в фрагмент рассматриваемого примера.

3.3. Постановка задачи

Задача формирования оптимальной организационной структуры управления рассматривается в плане выбора рационального уровня иерархии управления для реализации определенного набора функции управления и рационального разбиения структурного подразделения на отдельные группы при

управления (вершины графа), и связи между функциями для достижения конечной цели управления (ребра графа). Веса ребер отражают степень взаимосвязи между функциями в процессе функционирования системы. В зависимости от цели системы управления для одного и того же состава функций меняются связи между ними, и в конечном итоге каждая из целей определяет свой рациональный вариант оптимальной иерархии структуры. Задача заключается в определении такой иерархии системы управления, которая бы сохранила приемлемую эффективность для диапазона целей управления системы в целом.

Следующей задачей после определения рациональной иерархии системы управления является разбиение каждой из структурных единиц на самостоятельные группы, реализующие конкретные задачи функции, за которую несет ответственность данная структурная единица. Как и в первом случае, связь между задачами зависит от конечной цели системы управления. Целью моделирования на этом этапе является нахождение такого разбиения структурной единицы на самостоятельные группы, которое бы обеспечивало приемлемую эффективность решения задач при изменении связей между ними в определенных пределах.

3.4. Порядок выполнения работы

Лабораторная работа реализована в виде программного продукта, выполняемого в вычислительной среде WINDOWS 95. В соответствии с пунктами меню формируются матрицы взаимосвязей между структурными единицами системы управления или между самостоятельными группами структурных единиц. Количество матриц (количество целей управления) задается пользователем. В процессе работы программа генерирует для каждой таблицы оптимальный граф и есть возможность получения всех возможных вариантов перестройки исходного графа. Программа позволяет проанализировать полученные графы для всех матриц и выбрать рациональный вариант организационной структуры на множестве матриц.

3.5. Содержание отчета

1. Цель работы.
2. Исходные данные для расчета: число вершин исходного графа, матрицы взаимосвязей между вершинами для каждой из целей системы управления, условия решения задачи.
3. Для каждой матрицы оптимальный вариант графа и два-три варианта графа с характеристиками, близкими к оптимальным.
4. Окончательный вариант графа организационной структуры в заданных условиях решения задачи.
5. Выводы.

3.7. Контрольные вопросы

4. Окончательный вариант графа организационной структуры в заданных условиях решения задачи.

5. Выводы.

3.7. Контрольные вопросы

1. Основные задачи при формировании оптимальной организационной структуры управления.

2. Методы определения оптимальной иерархии организационной структуры управления и оптимального разбиения структурных единиц на самостоятельные группы.

3. Порядок выбора рационального варианта организационной структуры в условиях неопределенности целей управления системы.

Лабораторная работа №4

Планирование вклада средств в производство

4.1. Цель работы

1. Ознакомление с алгоритмом динамического программирования при планировании многоэтапных операций.
2. Планирование вклада средств в производство с целью получения максимального дохода.

4.2. Общие сведения

Задача о планировании вклада средств в производство с целью получения максимального дохода относится к типичной многоэтапной операции, которую можно планировать с помощью аппарата динамического программирования [3, 4]. При этом управление на каждом шаге выбирают с учетом будущего, а само такое управление называют условным оптимальным управлением. Алгоритм динамического программирования включает следующие пункты.

1. На выбранном шаге задаем набор (определяется условиями-ограничениями) значений переменной, характеризующей последний шаг, возможные состояния системы на предпоследнем шаге. Для каждого возможного состояния и каждого значения выбранной переменной вычисляем значения целевой функции. Из них для каждого исхода предпоследнего шага выбираем оптимальные значения целевой функции и соответствующие им значения рассматриваемой переменной. Для каждого исхода предпоследнего шага запоминаем оптимальное(ые) значение(я) переменной и соответствующее значение целевой функции. Получаем таблицу.

2. Переходим к оптимизации на следующем шаге при любом значении переменной и при оптимальных значениях следующих переменных. Оптимальное значение целевой функции на последующих шагах (при оптимальных значениях последующих переменных) считывают из предыдущей таблицы. Если новая переменная характеризует первый шаг, то переходим к п.3. В противном случае повторяем п.2 для следующей переменной.

3. При заданном в задаче исходном условии для каждого возможного значения первой переменной вычисляем значение целевой функции. Выбираем оптимальное значение целевой функции и соответствующее(ие) оптимальное(ые) значение(ия) первой переменной.

4. При известном оптимальном значении первой переменной определяем исходные данные для следующего(второго) шага и по последней таблице - оптимальное(ые) значение(ия) следующей (второй) переменной.

Рассмотрим на примере схему этой процедуры[5].

Планируется деятельность цехов (например, швейного и по ремонту обуви) сроком на 5 лет ($m=5$). Функции вклада в производство средств $\varphi(x) = 0,75x$, $\psi(y)$

Планируется деятельность цехов (например, швейного и по ремонту обуви) сроком на 5 лет ($m=5$). Функции вклада в производство средств $\langle p(x) = 0,75x, \psi(y) = 0,3y$, т.е. если средства z вкладывать только в производство первого цеха, то по годам средств будет вкладываться $z \cdot 0,75; z \cdot 0,75^2; z \cdot 0,75^3, \dots$. Соответственно, если средства z вкладывать только в выпуск продукции второго цеха, получим $g \cdot 0,3; z \cdot 0,3^2; z \cdot 0,3^3, \dots$.

Функции дохода как функции от объема вкладываемых средств x и y имеют вид

$$f(x) = 1 - e^{-x};$$

$$g(y) = 1 - e^{-2y}.$$

Требуется распределить имеющиеся ресурсы $g = 2$ между цехами по годам так, чтобы получить максимальный доход.

Решение задачи не удастся получить полностью в аналитическом виде, решим ее графически.

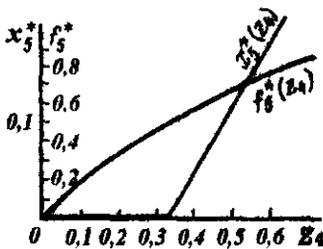


Рис.4.1. Зависимость f_5^* и x_5^* от z_4

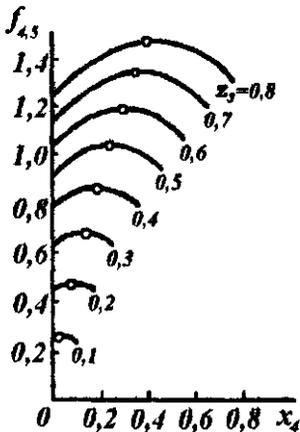


Рис.4.2. Зависимость $f_{4,5}$ от x_4

Пусть к началу пятого года количество средств равно z_4 и на пятый Год выделено средств первому цеху x_5 , тогда второму цеху будет выделено $y_5 (y_5 = z_4 - x_5)$. Чтобы найти условное оптимальное управление на пятом шаге $x_5(z_4)$, нужно для каждого z_4 найти максимум функции:

$$f_5 = 1 - e^{-x_5} + 1 - e^{-2(z_4 - x_5)} = 2 - [e^{-x_5} + e^{-2(z_4 - x_5)}].$$

При фиксированном z_4 максимум функции f_5 достигается либо при $x_5=0$, либо внутри отрезка $(0, z_4)$. Максимум внутри отрезка будет в точке x_5 , которую находят из условия

$$\frac{\partial f_5}{\partial x_5} = 0; e^{-x_5} - 2e^{-2(z_4 - x_5)} = 0;$$

То есть при $z_4 > \ln 2/2 = 0,347$ максимум достигается в точке

$$x_5^*(z_4) = 1/3(2z_4 - \ln 2).$$

При $z_4 \leq \ln 2/2$ максимум будет в точке $x_5^*(z_4) = 0$ Итак, условное оптимальное управление на пятом шаге

$$x_5^*(z_4) = \begin{cases} 0 & \text{при } z_4 \leq \ln 2/2; \\ 1/3(2z_4 - \ln 2) & \text{при } z_4 > \ln 2/2. \end{cases}$$

Построим на графике (рис.4.1) зависимость $x_5^* = x_5^*(z_4)$ и функции $g_5^* = f_5^*(z_4)$. При построении $f_5^*(z_4)$ пользуемся зависимостью (рис.4.2).

Перейдем к оптимизации на шаге 4: найдем оптимальные соотношения для вклада средств на шаге 4. Запас средств после шага 3

может изменяться z_3 . Наибольшее значение z_3 будет достигнуто, если все средства вложить в первый цех:

$$z_{3\max} = z * 0,75^3 = 2 * 0,75^3 = 0,844.$$

Наименьший запас средств соответствует случаю, когда все средства на трех шагах вкладывались во второй цех:

$$z_{3\min} = z * 0,3^3 = 0,054.$$

Будем рассматривать теперь набор значений z_3 от 0,1 до 0,8 с шагом 0,1 и для каждого значения z_3 найдем условное оптимальное управление на шаге 4 $x_4^*(z_3)$ и условный оптимальный доход на двух последних шагах $f_{45}^*(z_3)$

Для этого построим серию кривых, отражающих выигрыш на двух последних шагах при любом управлении на четвертом и при оптимальном на пятом:

$$f_{45} = f_4(z_3, x_4) + f_5^*(z_4) = f_4(z_3, x_4) + f_5^*[0,75x_4 + 0,3(z_3 - x_4)];$$

$$f_4(z_3, x_4) = 2 - [e^{-x_4} + e^{-2(z_3 - x_4)}].$$

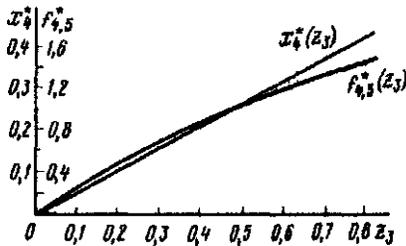


Рис.4.3. Зависимость f_{45}^* и x_4^* от z_3

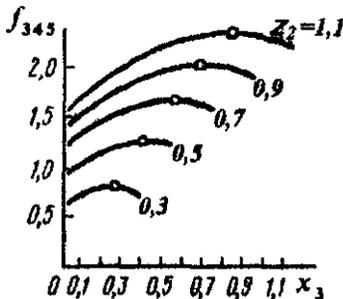


Рис. 4.4. Зависимость f_{345} от z_3

$f_5^*(z_4)$ берем из рис. 4.1.

Значения $f_{45}^*(z_4)$ приведены на рис.

4.2. Для каждого значения z_3 максимальная ордината определяет условный максимальный выигрыш на двух последних шагах $f_{45}^*(z_3)$ а абсцисса - условное оптимальное управление $x_4^*(z_3)$

На рис.4.3 даны графики зависимостей $f_{45}^*(z_3)$ и $x_4^*(z_3)$

Аналогично решают задачу для шагов 3 и 2. Результаты приведены на на рис. 4.4 - 4.7.

Спланируем шаг 1. При любом управлении на шаге 1 и оптимальном на последующих доход на пяти шагах

$$f_{12345}(x_1) = f_1(z, x_1) + f_{2345}^*(z_1) = 2 - [e^{-x_1} + e^{-2(z-x_1)}] + f_{2345}^*(z_1);$$

$$z_1 = 0,75x_1 + 0,3(z - x_1).$$

На шаге 1 имеющиеся ресурсы известны из условия задачи: $z = 2$. Поэтому $f_{12345}(x_1)$ можно описать одной

линией (рис. 4.8). Из рис.4.8 видно, что на шаге 1 максимальный выигрыш будет

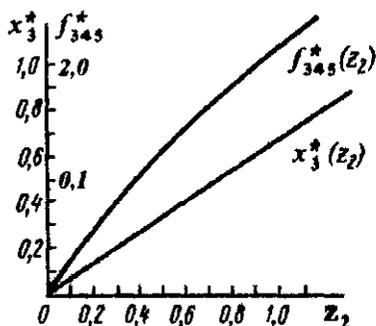


Рис.4.5. Зависимость f^*_{345} и x^*_{3} от z_2

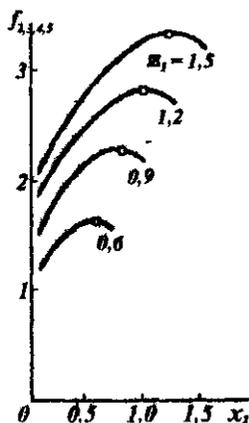


Рис.4.6. Зависимость $f_{1,3,4,5}$ от x_2

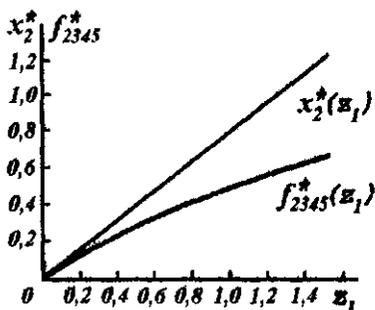


Рис.4.7. Зависимость f^*_{2345} и x^*_{2} от z_1

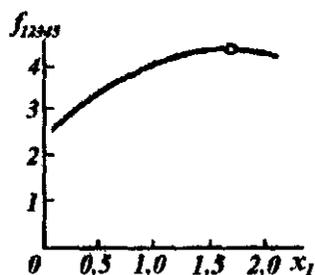


Рис.4.8. Зависимость f_{12345} от x_1

$f^*_{12345} = 4,35$; и соответствующее ему оптимальное управление (решение) $x^*_1 = 1,6$. По известному оптимальному управлению на первом шаге мы можем определить запас средств к концу этого шага, что определит оптимальное управление (решение) на следующем шаге.

Запас средств к концу шага 1

$$z^*_1 = 0,75x^*_1 + 0,3(z - x^*_1) = 1,32.$$

Тогда оптимальное управление на шаге 2 (рис 4.7) $x_2^* = 1,02$. Остаток средств к концу шага 2

$$z_3^* = 0,75x_2^* + 0,3(z_1^* - x_2^*) = 0,86.$$

Аналогично $x_3^*=0,62$; $z_3^*=0,54$; $x_4^*=0,30$; $z_4^*=0,30$; $x_5^*=0$.

Получим оптимальное управление, показывающее, какое количество средств надо вкладывать в первый цех:

$$x^* = \{x_1^*, x_2^*, x_3^*, x_4^*, x_5^*\} = \{1,6; 1,02; 0,62; 0,3; 0\}.$$

Найдем количество средств, вкладываемых во второй цех:

$$\begin{aligned} y_1^* &= z - x_1^* = 0,40; \\ y_2^* &= z_1^* - x_2^* = 0,30; \\ y_3^* &= z_2^* - x_3^* = 0,24; \\ y_4^* &= z_3^* - x_4^* = 0,24; \\ y_5^* &= z_4^* - x_5^* = 0,30; \end{aligned}$$

Определим остаток средств: $0,3 \cdot 0,3 = 0,09$.

4.3 Порядок выполнения работы

Задание содержит вид функций дохода для каждого из двух цехов, функции вклада средств в производство по всем годам 5-летнего периода планирования, объем распределяемого ресурса. В соответствии с инструкциями на каждом шаге пользователь определяет диапазон запаса средств $[z_{\min}, z_{\max}]$, величину шага разбиения диапазона, максимальные значения функции выигрыша с учетом последних и текущего шага. После оптимизации всех шагов находится оптимальное управление на каждом из них, содержащее объемы вкладываемых средств в первый и во второй цеха $X^* = \{x_1^*, x_2^*, x_3^*, x_4^*, x_5^*\}$ и $Y^* = \{y_1^*, y_2^*, y_3^*, y_4^*, y_5^*\}$, остаток средств, ожидаемый доход от вложенных средств. Сопоставляя остаток средств и ожидаемый доход, можно сделать вывод об целесообразности коррекции функции вклада средств в каждый из цехов и определить величину этой коррекции.

Лабораторная работа выполняется в следующей последовательности.

1. Задаются исходные данные для расчета: функции вклада средств в производство для каждого из двух цехов $\varphi(x)$ и $\psi(x)$, функции дохода $f(x)$ и $g(y)$, величину имеющегося ресурса z .

2. Оптимизация 5 шага с поиском условного оптимального управления $x_5(z_4)$.

3. Оптимизация 4 шага с поиском условного оптимального управления $x_4(z_3)$. При этом определяется $[z_3^{\max}, z_3^{\min}]$, шаг разбиения диапазона, в автоматическом режиме строится серия кривых $f_{45}(x_4)$. Пользователь с помощью курсора фиксирует локальные максимумы кривых $f_{45}(x_4)$ и автоматически строятся графики $f_{45}^*(z_3)$ и x_{43}^*

4. Аналогично решают задачу для шагов 3 и 2.

5. После планирования шага 1 и получения зависимости $f_{12345}(x_1)$ определяют абсциссу максимума графика и находят таким образом x_1^*

6. Последовательно определяются запасы средств z_1^* , после чего с помощью рис.4.7 - x_2^* , z_2^* с помощью рис.4.5 - x_3^* и т.д.

7. Получив $X^* = \{x_1^*, x_2^*, x_3^*, x_4^*, x_5^*\}$ и $Y^* = \{y_1^*, y_2^*, y_3^*, y_4^*, y_5^*\}$, остаток средств, ожидаемый доход от вложенных средств, определяется необходимость коррекции функций вклада по каждому из цехов. В случае необходимости коррекции возвращаются к пункту 1 и повторяют весь цикл вычислений с учетом новых функций вклада.

8. Проанализировать полученные результаты и составить отчет.

4.4. Содержание отчета

1. Исходные данные для решения задачи оптимизации вклада средств в производство.

2. Графики и расчеты функций условных ожидаемых доходов функционирования цехов по каждому году планируемого периода.

3. Окончательный вариант распределения вкладываемых средств по цехам и итоговая оценка полученного дохода.

4. Выводы.

4.5. Контрольные вопросы

1. Условия применимости динамического программирования.

2. Алгоритм динамического программирования.

3. Основные этапы метода планирования вклада средств в производство.

4. Как изменится процедура вычислений, если ввести переменные по годам функции дохода, функции вклада средств в производство.

Лабораторная работа №5

Методы решения задач календарного планирования

5.1. Цель работы

1. Ознакомление с процессом оперативного управления производством.
2. Ознакомление с моделями календарного планирования, упорядочиваемыми во времени работы или события.

5.2. Основные положения

5.2.1. Постановка общей задачи календарного планирования

Имеется множество деталей или партий деталей $\{Z_i\} = \{Z_1, Z_2, \dots, Z_i, \dots, Z_m\}$, подлежащих обработке; множество групп оборудования $\{B_j\} = \{B_1, B_2, \dots, B_j, \dots, B_n\}$, на которых предполагается обработка этих деталей, число единиц оборудования каждой группы $\{P_j\} = \{P_1, P_2, \dots, P_j, \dots, P_n\}$. Для каждой детали задана технология обработки детали $\{S_i\} = \{Z_{ij}, Z_{i1}, \dots, Z_{ik}\}$, $j, i, k \in n$, $i = 1, 2, \dots, m$, показывающая последовательность обработки i -й детали на станках $j, 1, \dots, k$, матрица длительности операций $\{t_i\} = \{t_{ij}, t_{i1}, \dots, t_{ik}\}$, $i = 1, 2, \dots, m$, где t_{ij} - время обработки i -й детали на j -м станке.

В качестве управляемых параметров выступает матрица расписания $G = \|t_{ij}\|$, $i = 1, 2, \dots, m$, $j = 1, 2, \dots, n$, где t_{ij} - время начала операции ij (обработка i -й детали на j -м станке).

Ограничения задачи

- 1) на одной единице оборудования не может обрабатываться больше одной детали;
- 2) каждая деталь в данный момент может обрабатываться только на одной единице оборудования;
- 3) если обработка детали на станке начинается, то она продолжается до конца операции;
- 4) последующая операция начинается после полного завершения предыдущей.

Критерий оптимальности имеет вид $T = \min_{ij} \max_{ij} t_{ij}$, где t_{ij} - время завершения работ по обработке всех деталей, t_{ij} - время завершения самой поздней операции ij .

Все методы решения задач календарного планирования можно разбить на три группы: точные, приближенные, эвристические [12]. К точным методам относятся, в частности, метод Джонсона, метод линейных диаграмм Ганта, метод Эйкера.

5.2.2. Метод Джонсона

Алгоритм метода Джонсона для 2-х станков.

1) $\min_{Z_i} t_{Z_i}^* = t^*$;

2) если $t^* = t_{Z_i,1}^*$ для какой-либо одной детали Z_i (в первом столбце находится один минимум), то деталь Z_i помещается в начало расписания;

3) если $t^* = t_{Z_i,2}^*$ для какой-либо одной детали Z_i (во втором столбце находится один минимум), то деталь Z_i помещается в конец расписания;

4) если $t^* = t_{Z_i,1}^*$ для 2-х и более деталей Z_i (в первом столбце находится несколько минимумов), то находится $\min\{t_{Z_i,2}, t_{Z_{i+1},2}\}$. Та деталь, для которой минимально время обработки на втором станке, ставится в конец расписания, а другая - в начало;

5) если $t^* = t_{Z_i,2}^*$ для 2-х и более деталей Z_i (во втором столбце находится несколько минимумов), то находится $\min\{t_{Z_i,1}, t_{Z_{i+1},1}\}$. Та деталь, для которой минимально время обработки на первом станке, ставится в начало расписания, а другая - в конец;

6) если $t^* = t_{Z_i,1}^* = t_{Z_i,2}^*$ для 2-х и более деталей Z_i (минимумы находятся в различных столбцах), то детали, у которых минимально время обработки на первом станке $t_{Z_i,1}^*$, ставятся в начало расписания, а детали, у которых минимально время обработки на втором станке $t_{Z_i,2}^*$, ставятся в конец;

7) из таблицы вычеркиваются детали, включенные в расписание на данном шаге. Процесс повторяется до тех пор, пока все детали не будут упорядочены.

На рис.5.1 приведена матрица длительности операций и пример построения по ней расписания методом Джонсона.

детали	станки	
	1	2
1	4	6
2	8	3
3	3	7
4	3	2
5	7	8
6	5	4

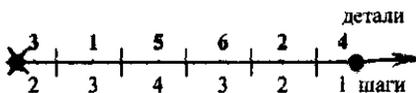


Рис.5.1. Пример Построения расписания методом Джонсона

5.2.3.Метод Эйкерса

Метод составляет расписание O работы станков, которые обрабатывают две детали $Z1$ и $Z2$ по заданному технологическому маршруту.

Метод Эйкера является графическим методом. На координатных осях наносятся времена t_{ij} обработки деталей Z_i в соответствии с технологическим маршрутом $\{S_i\}$. Полученная полуплоскость разбивается прямыми, проведенными параллельно осям через точки, соответствующие временам обработки t_{ij} . Прямоугольники с одноименными сторонами штрихуются. Задача построения оптимального расписания заключается в построении ломаной линии минимальной длины, соединяющей левый нижний и правый верхний углы полуплоскости и не заходящей в заштрихованные прямоугольники. Если ломаную можно провести только вертикально или горизонтально, то это соответствует тому, что одна деталь обрабатывается, а другая пролеживает, т.к. занят требуемый станок. Если ломаная идет по диагонали, то это соответствует тому, что обе детали обрабатываются одновременно.

Метод характеризуется простотой и наглядностью построения расписания. В то же время он применим лишь для построения расписания для двух-трех деталей и сравнительного анализа вариантов.

На рис.5.2 приведен пример построения расписания для двух деталей по методу Эйкера.

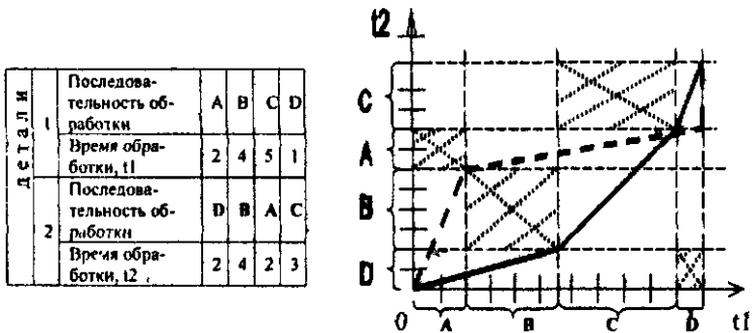


Рис.5.2. Пример построения расписаний методом Эйкера

5.2.4.Метод линейных диаграмм Ганта

Суть метода состоит в том, что формируются все возможные варианты расписания и для каждого из них строится диаграмма Ганта (рис.5.3).

Для построения дерева вариантов используется следующий алгоритм конфликтного перебора [10]. В момент окончания каждой операции ситуация может быть конфликтной и неконфликтной. Пусть в момент времени t на оборудование V_j поступило на обработку k_1 операций и k_2 операций выполняется, а k_3 ожидает обработки. Если $k_1+k_2+k_3 > P_j$, то ситуацию будем называть конфликтной в момент t на оборудовании V_j .

1. Анализируем первый столбец матрицы технологий. Выбираем элементы, соответствующие обработке на первом оборудовании. При этом возможны два случая: если $k_1 < P_1$ - неконфликтная ситуация, тогда все операции назначаем на обработку; если $k_1 > P_1$ - конфликтная ситуация. Конфликт имеет

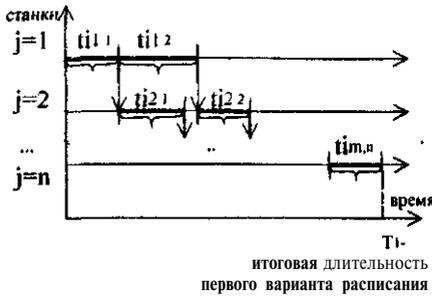


Рис 5 3 Пример гант-карты

$M_1 = C_{k_1}^{P_1} (k_1 - P_1)!$ способов разрешения. Выбираем из числа конфликтующих k операций для занесения на обработку, для остальных $k - P_1$ операций устанавливаем приоритет на обработку и выстраиваем их в очередь. При освобождении оборудования первой группы на обработку заносится первая операция в очереди.

Аналогично поочередно выбираем элементы, соответствующие обработке на других группах

оборудования, и назначаем для них операции на обработку.

В результате первого шага все оборудование делится на свободное $k_j < P_j$, занятое без очереди $k_j = P_j$; занятое и имеющее очередь $k_j > P_j$.

2. В результате окончания какой-либо операции освобождается единица оборудования из V_1 -й группы и появляется претендент на оборудование V_j -й группы. На V_j -й группе оборудования, которую требует Z_i деталь на 2-м шаге обработки, может возникнуть конфликтная ситуация.

Таким образом алгоритм сводится к назначению операций на соответствующие группы оборудования и разрешению конфликтных ситуаций.

Пример построения линейных диаграмм Ганта.

Имеется множество деталей или партий деталей $\{Z_i\} = \{Z_1, Z_2, Z_3\}$, подлежащих обработке; множество групп оборудования $\{B_j\} = \{B_1, B_2\}$, на кптоых поелполагается обработка этих деталей, число единиц оборудования

каждой группы $\{P_j\} = \{2, 1\}$. Задана технология обработки детали $|S_i| = \begin{vmatrix} 1 & 2 \\ 1 & 2 \\ 1 & 2 \end{vmatrix}$

матрица длительности операций $|T_i| = \begin{vmatrix} 1 & 3 \\ 9 & 5 \\ 3 & 2 \end{vmatrix}$. Требуется составить оптимальное

расписание.

Следуя приведенному выше алгоритму, сначала анализируем первый столбец матрицы технологий. Выбираем элементы, соответствующие обработке на первой группе оборудования. Так как таких элемента три, а число единиц оборудования первой группы - два, то возникает конфликт. Конфликт разрешим

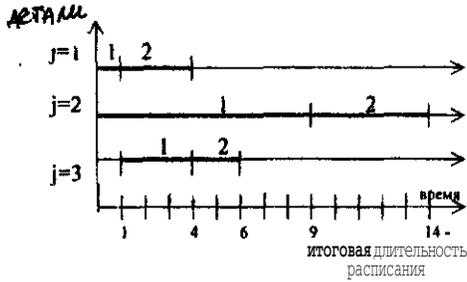


Рис.5.4. Гант-карта, построенная с учетом вектора приоритетов деталей $\{Pr\}=\{1,2,3\}$.

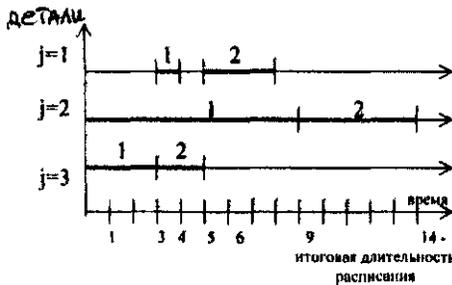


Рис.5.5. Гант-карта, построенная с учетом вектора приоритетов деталей $\{Pr\}=\{3,2,1\}$.

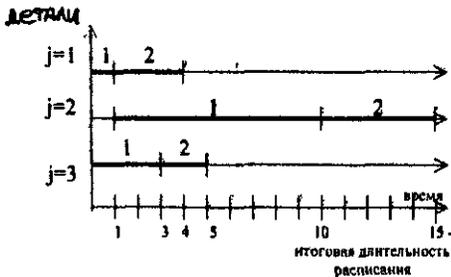


Рис.5.6. Гант-карта, построенная с учетом вектора приоритетов деталей $\{Pr\}=\{3,1,2\}$.

других векторов приоритетов разрешения конфликтных ситуаций (например, для приведенных на рис.5.5 и рис.5.6). Рассмотрев все возможные варианты,

$C_3^2(3-2)! = 3$ способами. Выбираем из числа конфликтующих, например, первую и вторую детали для занесения на обработку на первом станке, для остальных (в нашем случае - третьей детали) устанавливаем приоритет на обработку и выстраиваем их в очередь. При освобождении оборудования Первой группы на обработку заносится первая деталь из очереди. Первая и вторая детали после окончания обработки на первой группе оборудования требуют обработки на оборудовании второго типа. Поэтому на оборудовании второго типа также может возникнуть конфликтная ситуация, если на обработку на одной единице оборудования претендуют одновременно и первая и вторая детали. В нашем случае первой заканчивает обрабатываться первая деталь, назначим ее на обработку оборудованием второго типа, затем назначаем вторую деталь. В результате окончания обработки третьей детали освобождается единица оборудования из 1-й группы и появляется претендент на оборудование 2-й группы. На 2-й группе оборудования, которую требует третья деталь на 2-м шаге обработки, может возникнуть конфликтная ситуация, которая разрешается аналогичным образом. На рис.5.4 приведена гант-карта, иллюстрирующая этот пример, она наглядно отражает процесс построения расписания с учетом вектора приоритетов разрешения конфликтных ситуаций $\{Pr\}=\{1,2,3\}$. Аналогичным образом строятся гант-карты для

выбирают оптимальное расписание, которое соответствует наименьшему времени окончания последней операции(в нашем случае - Мед.).

Метод позволяет получить точное расписание и учесть приоритет обработки деталей. В то же время с увеличением размерности задачи возникает проблема перебора всех возможных вариантов расписания.

5.3. Порядок выполнения работы

Лабораторная работа выполняется в следующей последовательности:

1. В соответствии с заданными исходными данными с помощью профаммы, реализующей метод Джонсона, составляется оптимальное расписание как для 2-х, так и для 4-х станков. Ручным расчетом составляется оптимальное расписание и сравнивается с результатами работы профаммы.

2. При работе с методом Эйкерса задают диапазоны времен обработки деталей на каждом из станков. Варьируя значениями времен из диапазонов, находят расписание, удовлетворяющее заданной продолжительности полной обработки этих деталей. При этом определяются критические и некритические операции с точки зрения выполнения обработки деталей за контрольное время.

3. Строятся линейные диаграммы Ганта с учетом и без учета приоритетов при обработке деталей. Определяется приоритет, позволяющий получить расписание, близкое к оптимальному, и при этом избежать полного перебора всех возможных вариантов расписания.

5.4.Содержание отчета

1. Исходные данные для составления расписания по каждому из методов.
2. Очередность обработки деталей, полученная методом Джонсона.
3. Гант-карта расписания обработки деталей.
4. График расписания обработки деталей, полученный с помощью метода Эйкерса.
5. Выводы.

5.5. Контрольные вопросы

1. Общая постановка задачи календарного планирования.
2. Основные этапы методов Джонсона, Эйкерса, Ганта.
3. Проведите сравнительный анализ достоинств и недостатков указанных методов решения задач календарного планирования.
4. В чем суть точных методов составления расписания: фациентного метода, метода ветвей и границ.
5. Приведите примеры приближенных и эвристических методов составления расписания.

Лабораторная работа № 6

Оптимальное отображение множества функций системы по ее элементам

6.1. Цель работы

Изучение частных задач отображения функций системы по элементам системы.

6.2. Общие сведения

На этапе общесистемного проектирования одной из главнейших задач является задача проектирования структуры.

Применительно к человеко-машинным системам наиболее распространенным является случай, когда принципы построения системы, выполняемые функции и компоненты системы, оказываются заданными [7]. Тогда задача синтеза оптимальной структуры заключается в определении оптимального отображения множества выполняемых функций на множество взаимосвязанных элементов, учитывая ограничения на ресурсы, на техническую реализацию, на загрузку технических средств, на своевременность решения задач. Т.е. необходимо найти оптимальный состав узлов системы и связей между ними, возлагаемые на технические средства задачи и варианты их решения, распределение их по уровням и узлам системы и выбор комплекса технических средств, при которых максимизируется эффект от внедрения задач.

Поскольку множество возможных комбинаций конечно, теоретически существует возможность отыскания оптимального варианта распределения. В действительности реализовать эту возможность довольно затруднительно, так как в реальных задачах количество узлов системы, количество вариантов решения задач обычно велико, из-за наличия ограничений алгоритмы формирования комбинаций и оценки этих распределений достаточно сложны, эффективность системы является функцией очень большого количества факторов.

В настоящее время не существует единого метода решения экстремальных задач подобного типа. Сложность решения задачи синтеза оптимальной структуры приводит к тому, что на практике ставят и решают более частные задачи синтеза, например, такие, как определение оптимального распределения возлагаемых на систему функций по заданным уровням и узлам системы, определение оптимальных вариантов реализации функций в системе, выбор комплекса технических средств, обеспечивающего качественную реализацию функций, и т.д.

Рассмотрим некоторые частные постановки задач формализованного распределения множества решаемых задач между различными узлами системы при различных критериях и ограничениях [6, 8].

6.3. Первая частная задача синтеза оптимальной структуры системы

Необходимо так распределить i задач, $i = 1, I$, между j узлами, $j = 1, J$, чтобы обеспечить минимум общих затрат или минимум общего времени решения при выполнении ограничений на загрузку каждого из узлов, или на затраты в каждом j -м узле.

Математическая модель этой задачи может быть записана следующим образом: найти

$$\min \sum_{i=1}^I \sum_{j=1}^J a_{ij} x_{ij} \quad (6.1)$$

при

$$\sum_{i=1}^I a_{ij} x_{ij} \leq b_j, \quad j = \overline{1, J}; \quad (6.2)$$

$$\sum_{j=1}^J x_{ij} = 1, \quad i = \overline{1, I}; \quad (6.3)$$

$$x_{ij} = \begin{cases} 0 \\ 1 \end{cases}.$$

В этих соотношениях a_{ij} -затраты (время решения) i -й задачи в j -м узле; b_j -допустимые затраты (загрузка) в j -м узле; переменная $x_{ij}=1$, если i -я задача решается в j -м узле, и $x_{ij}=0$ -- в противном случае; условие (6.3) означает, что каждая задача должна решаться только в одном узле

Итак, заданы матрица $\|a_{ij}\|$ и вектор $\|b_j\|$

Наиболее удобным для решения данного класса задач является метод "ветвей и границ". Применительно к данной задаче он заключается в направленном движении по вершинам дерева, полученного путем фиксирования части переменных $x_{ij}(x_{ij}=0$ или 1).

Вершины первого уровня получают, поочередно закрепляя для $j=1, 2, 3, \dots$ при $i=1$. первую задачу за первым узлом, вторым и т.д., т.е. фиксируя $x_{ij}'=1$

Вершины второго уровня получают, фиксируя x_{ij}'' для $j=1, 2, 3, \dots$ при $i=2$ и т.д.

Для каждой вершины находят оценку

$$\sum_{i < i^*} a_{ij} + \sum_{i > i^*} \overline{a_{ij}}, \quad (6.4)$$

где i^* - число рассмотренных уровней ветвления

$$a_{ij} = \min_j a_{ij}$$

Стратегия ветвления может быть улучшена за счет использования специфических свойств рассматриваемой задачи, что существенно при решении задач большой размерности.

Вначале из матрицы коэффициентов $\|a_{ij}\|$ системы (6.1) исключаем все элементы, для которых выполняется условие $a_{ij} > b_j$, $i = \overline{1, I}$, $j = \overline{1, J}$. При этом возможны следующие случаи, когда в некоторой строчке:

исключены все элементы a_{ij} , тогда решение отсутствует;

остался лишь один элемент a_{ij} , он обязательно входит в оптимальное решение, если оно существует; значение b_j заменяется на $b_j - a_{ij}$ и этот элемент в дальнейшем поиске не участвует;

осталось несколько элементов, они участвуют в дальнейшем поиске оптимального решения.

Пример решения.

Найти $\min \sum_{i=1}^4 \sum_{j=1}^4 a_{ij} x_{ij}$ при ограничениях

$$\sum_{i=1}^4 a_{ij} x_{ij} \leq b_j, \quad \sum_{j=1}^4 x_{ij} = 1, \quad x_{ij} = 1 \vee 0;$$

$$a_{ij} = \begin{vmatrix} 8 & 5 & 6 & 4 \\ 5 & 3 & 2 & 2 \\ 1 & 7 & 3 & 4 \\ 2 & 3 & 8 & 5 \end{vmatrix}; \quad b_j = \{3 \quad 2 \quad 5 \quad 3\}$$

1. В соответствии с ранее рассмотренным алгоритмом производим упрощение $\|a_{ij}\|$, для чего исключаем элементы, для которых выполняется условие $a_{ij} > b_j$. Первая строчка после исключения не содержит ни одного элемента, т.е. первая задача не может быть решена, значит решение отсутствует.

2. Пусть $b_j = \{3 \quad 6 \quad 5 \quad 3\}$. Тогда после соответствующего исключения $\|a_{ij}\|$ имеет вид

$$a_{ij} = \begin{vmatrix} - & 5 & - & - \\ - & 3 & 2 & 2 \\ 1 & - & 3 & - \\ 2 & 3 & - & - \end{vmatrix}.$$

Первая строчка содержит только один элемент $a_{12} = 5$, следовательно, он обязательно войдет в решение. Итак, $x_{12} = 1$. Ресурс на второй узел 6, следовательно, остается резерв $b_2 - a_{12} = 6 - 5 = 1$, теперь $b_j = \{3 \quad 1 \quad 5 \quad 3\}$.

3. Далее процедура аналогична рассмотренной выше, но каждый раз ищем минимальные элементы в столбцах и проверяем, не перегружен ли данный узел.

$$a_{ij} = \begin{vmatrix} - & - & 2 & 2 \\ 1 & - & 3 & - \\ 2 & - & - & - \end{vmatrix} \quad b_j = \begin{vmatrix} 3 & 1 & 5 & 3 \end{vmatrix}$$

4. Выбираем минимальные элементы в каждой строке

$$a_{ij} = \begin{vmatrix} 1 & - & 3 & - \\ 2 & - & - & - \end{vmatrix}, \quad b_j = \begin{vmatrix} 3 & 1 & 3 & 3 \end{vmatrix} \vee b_j = \begin{vmatrix} 3 & 1 & 5 & 1 \end{vmatrix}$$

5. $a_{ij} = \begin{vmatrix} 2 & - & - & - \end{vmatrix}, \quad b_j = \begin{vmatrix} 2 & 1 & 3 & 3 \end{vmatrix} \vee b_j = \begin{vmatrix} 2 & 1 & 5 & 1 \end{vmatrix}, \quad x_{23} = 1 \vee x_{21} = 1, \quad x_{31} = 1$

6. Загрузка не превышает заданную. Окончательно

$$x_{ij} = \begin{vmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{vmatrix} \vee x_{ij}' = \begin{vmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{vmatrix}$$

Значение целевой функции в первом случае $(x_{ij}) \quad 5+2+1+2=10$, во втором $(x_{ij}') \quad 5+2+1+2=10$.

6.4. Вторая частная задача синтеза оптимальной структуры

Необходимо так распределить i задач, $i = \overline{1, I}$, между j узлами, $j = \overline{1, J}$, чтобы обеспечить минимум общих затрат или минимум общего времени решения при выполнении ограничений на общее время решения или общие затраты соответственно.

Математическая модель этой задачи может быть записана следующим образом: найти

$$\min \sum_{i=1}^I \sum_{j=1}^J a_{ij} x_{ij} \quad (6.5)$$

при

$$\sum_{i=1}^I \sum_{j=1}^J b_{ij} x_{ij} \leq B, \quad j = \overline{1, J}; \quad (6.6)$$

$$\sum_{j=1}^J x_{ij} = 1, \quad i = \overline{1, I}; \quad (6.7)$$

$$x_{ij} = \begin{cases} 0 \\ 1 \end{cases}$$

В этих соотношениях a_{ij} -затраты (время решения) i -й задачи в j -м узле; b_{ij} -время решения (затраты) i -й задачи в j -м узле; B - общее время решения

(затраты) всех задач; переменная $x_{jj}=1$, если 1-я задача решается в j -м узле, и $x_{jj}=0$ - в противном случае.

Для решения этой задачи прежде всего берутся минимальные элементы в каждой строке матрицы коэффициентов $\|a_{ij}\|$ и проверяется выполнение условия (6) для соответствующих элементов матрицы коэффициентов $\|b_{ij}\|$

Если условие (6.6) выполняется, то это и будет оптимальным решением. Если условие (6.6) не выполняется, то из матрицы коэффициентов $\|a_{ij}\|$ и $\|b_{ij}\|$ исключают те элементы, которые не могут войти ни в одно допустимое решение. Для этого последовательно рассматриваются все элементы матрицы $\|b_{ij}\|$ и проверяется условие

$$\sum_{i=1}^{\bar{I}} b_{ij} + b_{lj} + \sum_{i=1, i \neq l}^{\bar{I}} b_{ij} < B, \quad l = \bar{I}, \bar{I}, \quad j = \bar{I}, \bar{J}, \quad (6.8)$$

где \bar{b}_{ij} - минимальный элемент в соответствующей строке; b_{lj} - рассматриваемый элемент, $j \neq l$

Иначе говоря, каждая из задач последовательно закрепляется за каждым из узлов и проверяется выполнение условия (6.6) в лучшем случае.

Если (6.8) нарушается, то соответствующий элемент b_{lj} не входит в допустимое решение и он исключается из матрицы $\|b_{ij}\|$. Из матрицы $\|a_{ij}\|$ исключается соответствующий элемент a_{lj} .

Из условия (6.7) следует, что в каждой строке может быть только один элемент, поэтому $\min \sum_{i=1}^{\bar{I}} \sum_{j=1}^{\bar{J}} a_{ij} x_{ij}$ без учета выражения (6.6) равен $\sum_{i=1}^{\bar{I}} \min \sum_{j=1}^{\bar{J}} a_{ij} x_{ij}$

Отсюда, если для элементов одновременно выполняется условие

$$a_{ij} > a_{lj} \text{ и } b_{ij} \geq b_{lj} \quad (l \neq j) \quad (6.9)$$

то эти элементы могут быть исключены из рассмотрения.

Хотя исключение элементов не всегда приводит к оптимальному решению, объем вычислений резко сокращается.

Далее используется метод "ветвей и границ". В отличие от предыдущей задачи ветвление осуществляется с учетом ограничения (6.6), что существенно сокращает число рассматриваемых вариантов. Оценка для каждой вершины находится по элементам матрицы (6.5) аналогично предыдущей задаче (6.4). Ограничение при этом имеет вид

$$\sum_{i=1}^{\bar{I}} b_{ij} + \sum_{i=1}^{\bar{I}} \bar{b}_{ij}, \quad (6.10)$$

где l^* -уровень ветвления; $\bar{b}_{ij} = \min_j b_{ij}$

Рассмотрим числовой пример решения задач минимизации общих затрат при ограничениях на общее время решения. Пусть

$$a_{ij} = \begin{vmatrix} 3 & 7 & 2 & 4 \\ 4 & 8 & 1 & 3 \\ 5 & 9 & 6 & 2 \\ 6 & 10 & 7 & 1 \\ 7 & 5 & 3 & 1 \end{vmatrix}; \quad b_{ij} = \begin{vmatrix} 1,5 & 3 & 2 & 9 \\ 2 & 6 & 5 & 10 \\ 3 & 7 & 6 & 11 \\ 4 & 8 & 7 & 12 \\ 4 & 9 & 8 & 5 \end{vmatrix}; \quad B \leq 20.$$

1. Сначала находим минимальные элементы в каждой строке матрицы $\|a_{ij}\|$ и проверяем, удовлетворяется ли условие (6.6) по одноименным элементам матрицы $\|b_{ij}\|$:

$$\sum_{i=1}^5 b_{ij} = 2+5+11+12+5 > 20.$$

Условие не выполняется, и задачу сразу решить не удастся.

2 Приступим к упрощению матрицы. Для матрицы $\|b_{ij}\|$ последовательно для всех элементов проверяется условие (6.8):

$$i=1; \begin{cases} 1,5 + \underbrace{2+3+4+4}_{13} = 14,5 \\ 3+13 = 16 \\ 2+13 = 15 \\ 9+13 = 22 \end{cases} \quad i=2; \begin{cases} 2 + \underbrace{1,5+3+4+4}_{12,5} = 14,5 \\ 6+12,5 = 18,5 \\ 5+12,5 = 17,5 \\ 10+12,5 = 22,5 \end{cases}$$

и т.д., $i=3, 4, 5$.

Элементы b_{14} , b_{24} , b_{34} , b_{44} не удовлетворяют условию (6.8), они исключаются из матрицы $\|b_{ij}\|$ и одноименные элементы исключаются из матрицы $\|a_{ij}\|$:

$$a_{ij} = \begin{vmatrix} 3 & 7 & 2 & - \\ 4 & 8 & 1 & - \\ 5 & 9 & 6 & - \\ 6 & 10 & 7 & - \\ 7 & 5 & 3 & 1 \end{vmatrix}; \quad b_{ij} = \begin{vmatrix} 1,5 & 3 & 2 & - \\ 2 & 6 & 5 & - \\ 3 & 7 & 6 & - \\ 4 & 8 & 7 & - \\ 4 & 9 & 8 & 5 \end{vmatrix};$$

Находим минимальные элементы в каждой строке матрицы $\|a_{ij}\|$ и проверяем, удовлетворяется ли условие (6.6) по одноименным элементам матрицы $\|b_{ij}\|$

$$\sum_{i=1}^5 b_{ij} = 2+5+3+4+5 < 20. \quad \text{Следовательно, найдено оптимальное решение.}$$

Если решение не найдено, следует продолжить процесс упрощения матриц.

3. Т.к. в каждой строчке может быть только один элемент и в обеих матрицах осуществляется минимизация, то если одновременно выполняются

условия (6.9), то эти элементы могут быть исключены из рассмотрения. После соответствующих упрощений матрицы имеют вид

$$a_{ij} = \begin{pmatrix} 3 & - & 2 & - \\ 4 & - & 1 & - \\ 5 & - & - & - \\ 6 & - & - & - \\ 7 & 5 & 3 & 1 \end{pmatrix}; \quad b_{ij} = \begin{pmatrix} 1,5 & - & 2 & - \\ 2 & - & 5 & - \\ 3 & - & - & - \\ 4 & - & - & - \\ 4 & 9 & 8 & 5 \end{pmatrix}.$$

Условие (6.6) выполняется. Если бы ответ найти не удалось, пришлось бы вести ветвление и каждый минимальный вариант проверять на условие (6.6).

Ответ:

$$x_{ij} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

6.5. Третья частная задача синтеза оптимальной структуры

Необходимо так распределить i задач, $i = \overline{1, I}$, между j узлами, $j = \overline{1, J}$, чтобы обеспечить минимум общих затрат или минимум общего времени решения при выполнении ограничений на общее время решения и загрузку узлов, либо на общие затраты и загрузку узлов соответственно.

Математическая модель этой задачи может быть записана следующим образом: найти

$$\min \sum_{i=1}^I \sum_{j=1}^J a_{ij} x_{ij} \quad (6.10)$$

при

$$\sum_{i=1}^I \sum_{j=1}^J b_{ij} x_{ij} \leq B,$$

$$\sum_{j=1}^J c_{ij} x_{ij} \leq c_j, \quad j = \overline{1, J}; \quad (6.11)$$

$$\sum_{j=1}^J x_{ij} = 1, \quad i = \overline{1, I}; \quad (6.12)$$

$$x_{ij} = \begin{cases} 0 \\ 1 \end{cases}.$$

Для решения этой задачи прежде всего из матриц коэффициентов $\|a_{ij}\|$, $\|b_{ij}\|$ и $\|c_{ij}\|$ исключаются элементы, которые заведомо не могут войти в

оптимальное решение. Исключение элементов b_u и c_u из матриц систем (6.11) и (6.12) осуществляется аналогично рассмотренному выше примеру, т.е. исключаются все элементы, для которых не выполняется условие типа (6.8). Оценка для матрицы (6.10) находится аналогично оценке в первой задаче.

6.6. Порядок выполнения работы

Лабораторная работа выполняется в следующей последовательности

1. В соответствии с полученным заданием каждым из методов получить пользовательский вариант распределения функций по узлам. Оценить эффективность такого распределения.

2. С помощью программ продублировать решение задачи и сравнить полученные результаты с пользовательским вариантом.

3. Сравнить результаты отображения функций по узлам по каждой из методик и оценить эффективность каждой методики.

6.5. Содержание отчета

1. Исходные данные.

2. Варианты решения трех частных задач синтеза структуры АС, полученных с помощью ручных расчетов.

3. Варианты решения трех частных задач синтеза структуры АС, полученных с помощью программного пакета.

4. Выводы.

6.6. Контрольные вопросы

1. Перечислите основные задачи синтеза оптимальной структуры организационно-технических систем.

2. Постановка и метод решения первой частной задачи синтеза оптимальной структуры.

3. Постановка и метод решения второй частной задачи синтеза оптимальной структуры.

4. Постановка и метод решения третьей частной задачи синтеза оптимальной структуры.

ЛАБОРАТОРНАЯ РАБОТА № 7

АНАЛИЗ И РЕШЕНИЕ ЗАДАЧ РАСПРЕДЕЛЕНИЯ РЕСУРСОВ

7.1. Цель работы

1. Ознакомление с алгоритмами решения задач на графах.
2. Программная реализация алгоритма решения задачи оптимальной нагрузки графа.

7.2. Общие сведения

В современном мире существует множество реальных систем, таких, как электрическая сеть, железнодорожная система, коммуникационная сеть и любая другая распределенная система, для расчета которых используется аппарат теории графов. Первые задачи теории графов были связаны с решением математических развлекательных задач и головоломок, однако со временем они выросли в серьезные научные исследования. Так, попытки найти решение задачи о кенигсбергских мостах привели к возникновению задач об обходах графов; решение задачи о перевозках дало мощный толчок к созданию эффективных методов решения транспортных задач; попытки решить сформулированную в середине XIX века задачу четырех красок стали причиной целенаправленных исследований графов, имеющих теоретическое и прикладное значение.

В данной работе рассматривается реальная практическая задача с использованием взвешенного графа. Задача может быть сформулирована по-разному, а ее решения имеют широкую область применения.

Предположим, что имеется система водопроводных труб (рис.7.1,а). Каждая дуга представляет собой трубу, а число над дугой - пропускную способность данной трубы. Узлы являются местами соединения труб. Узлы S и F есть соответственно источник воды и сток. Вода из узла S должна быть передана по водопроводной системе в узел F. Вода течет по водопроводной системе только в одном направлении, причем в узел S не входит, а из узла F не выходит ни одна труба. Необходимо максимизировать объем воды, протекающий от источника к

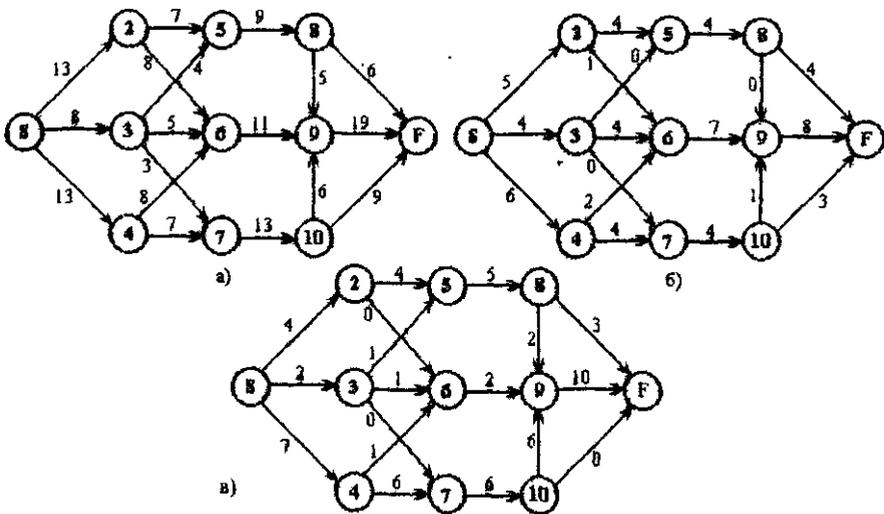


Рис.7.1. Задача распределения ресурсов на графе:
 а - функция пропускной способности; б, в - две возможные функции потоков

стоку. Взвешенный ориентированный граф (рис.7.1,а) представляет идеальную структуру данных для моделировании подобной ситуации.

Хотя источник может производить большой объем воды, а сток - потреблять примерно такой же объем, водопроводная система может не обеспечивать достаточную пропускную способность. Стало быть, критическим фактором всей системы является пропускная способность водопровода.

Функция пропускной способности $c(a,b)$ для двух узлов а и б равна пропускной способности трубы между этими Двумя узлами, если таковая имеется, и 0 в Противном случае. Через каждую трубу протекает определенный объем воды (может быть нулевой). Функция потока $f(a,b)$ для двух узлов а и б равна 0, если узел а не смежен с узлом б, и объему воды, протекающему по трубе из узла а в узел б, иначе. Ясно, что $f(a,b) > 0$ для любых узлов а и б. Более того, для любых узлов а и б выполняется соотношение $f(a,b) < c(a,b)$, поскольку по трубе не может протекать воды больше, чем обеспечивает ее пропускная способность. Пусть v обозначает объем воды, который проходит по системе труб от узла S к узлу F. Тогда общий объем воды, вытекающей из узла S по всем трубам, равен общему объему воды, поступающей в узел F через все трубы, и оба этих значения равны

v. Это может быть выражено равенством

$$\text{Sum}\{f(S,x)\} = \text{Sum}\{f(y,F)\} = v,$$

где x и y узлы соответственно смежные с истоком и со стоком.

Объем воды, вытекающей из любого узла, не совпадающего с узлами S и F, равен объему воды, поступающей в этот узел:

$$\text{Sum}\{(x,y)\} = \text{Sum}\{f(y,z)\},$$

где y - узел, не являющийся ни узлом S, ни узлом F.

Для данного графа и данной функции пропускной способности (см. рис.7.1,а) может существовать несколько функций Потока. На рис.7.1,б и 7.1,в приведены две возможные функции потока.

В результате решения необходимо найти функцию потока, максимизирующую значение v - объема воды, протекающего из узла S в узел P. Такая функция потока называется оптимальной. Ясно, что функция потока на рис.7.1,б лучше, чем функция потока на рис.7.1,в, поскольку для первой из них v равно 15, а для второй равно 13. Но нельзя, однако, до проведения вычислений утверждать, что функция потока на рис.7.1,б является оптимальной.

Стратегия нахождения оптимальной функции потока такова: начать с нулевой функции потока и последовательно улучшать ее до тех пор, пока не будет достигнута оптимальная функция потока.

При заданной функции потока существует два пути ее улучшения. Во-первых, можно стараться найти такой путь $S \rightarrow x_1 \rightarrow x_2 \rightarrow \dots \rightarrow x_n \rightarrow F$ из узла S в узел F, что поток по каждой дуге пути строго меньше ее пропускной способности, т.е. $f(x_{i-1},x_i) < c(x_{i-1},x_i)$ для всех $i=[2,n]$. Поток может быть увеличен на каждой дуге такого пути на величину, равную минимальному значению выражения $c(x_{i-1},x_i) - f(x_{i-1},x_i)$ для всех $i=[2,n]$.

Такой метод может быть проиллюстрирован графом на рис.7.2,а, где для

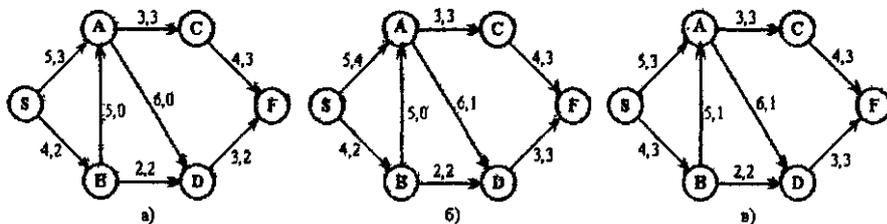


Рис.7.2. Пути улучшения функции потока: а - начальная функция потока; б, в - два возможных улучшения функции потока

каждой дуги указаны соответственно пропускная способность и текущий поток. Существует два пути из узла S в узел F с положительным значением потока: (S, A, C, F) и (S, B, D, F). Однако каждый из них содержит одну дугу ([A, C] и [B, D]), в которой поток равен пропускной способности. Стало быть, поток по этим путям не может быть улучшен. В то же время путь (S, A, D, F) таков, что пропускная способность каждой дуги пути больше, чем текущий поток. Максимальная величина, на которую может быть увеличен поток по этому пути, равна 1, поскольку поток по дуге [D, F] не может превышать 3. Получившаяся в результате функция потока показана на рис.7.2,б. Общий поток из узла S в узел F теперь увеличен с 5 до 6. На рис.7.2,в показана функция потока, получившаяся в результате улучшения пути (S, B, A, D, F) исходного графа. Эта функция также обеспечивает величину суммарного потока из узла S в узел F, равную 6, и, следовательно, не лучше и не хуже функции потока на рис.7.2,б.

Даже если не существует пути, поток по которому может быть улучшен, можно найти другой метод улучшения суммарного потока от источника к стоку. Метод проиллюстрирован на рис.7.3. На рис.7.3,а не существует пути из узла S в узел F, поток по которому может быть улучшен. Но если уменьшить поток из узла

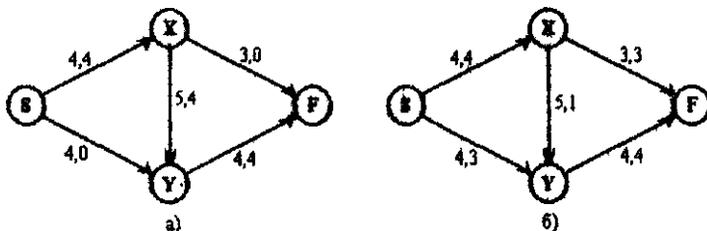


Рис.7.3. Пути улучшения функции потока: а-начальная функция потока; б - улучшенная функция потока

X в узел Y, то может быть увеличен поток из узла X в узел F. Для компенсации уменьшения входного потока в узел Y может быть увеличен поток из узла S в узел Y. Результатом этих изменений является увеличение общего потока из узла S в узел F. Поиск из узла X в узел Y может быть перенаправлен в узел F так, как показано на рис.7.3,б, и суммарный поток из узла S в узел F может, следовательно, быть увеличен с 4 до 7.

Второй метод можно обобщить следующим образом. Предположим, что существуют: путь из узла S в некоторый узел y, путь из некоторого узла x в узел F

и путь из узла x в узел y с положительным потоком. Тогда поток по пути из узла x в узел y может быть уменьшен, а потоки из узла x в узел F и из узла S в узел y могут быть увеличены на ту же величину. Величина изменения равна минимуму из потока из узла x в узел y и разностей между пропускной способностью и потоками на путях из узла S в узел y и из узла x в узел F :

$$\min (f(x,y), c(S,y) - f(S,y), c(x,F) - f(x,F)).$$

7.3. Алгоритм оптимизации потока

Пусть дан взвешенный граф (матрица пропускной способности $Weight$) с источником S и стоком F . Алгоритм нахождения оптимальной функции потока для этого графа может быть описан так:

- 1) инициализировать функцию потока $flow$ для каждой дуги в 0 ;
- 2) попытаться найти такой путь из S в F , который увеличивает поток в F на $plan > 0$;
- 3) если такой путь не может быть найден, то вернуть $plan$;
- 4) увеличить поток по каждой дуге пути на величину $plan$;
- 5) перейти к шагу 2.

Ядро алгоритма - шаг 2. Рассмотрим рекурсивный алгоритм нахождения пути более подробно:

1. установить переменную $plan$ (коррекция плана на текущем шаге) в $0xffff$ (максимально возможное значение);
2. установить указатель текущей вершины в 1 ;
3. установить указатель текущей дуги для текущей вершины в 1 ;
4. определить, есть ли связь по дуге: есть - пункт 5, нет - пункт 6;
5. определить, может ли быть увеличен поток по текущей дуге; да - пункт 8, нет - пункт 6;
6. увеличить счетчик текущей дуги и перейти к пункту 4, в случае, если не все дуги просмотрены; если же у текущей вершины смежных вершин больше нет перейти к пункту 7;
7. вернуть 0 ;
8. присвоить указателю текущей вершины номер конечного узла рассматриваемой дуги;
9. подкорректировать значение переменной $plan$:
 $plan = \text{Min}(plan, flow)$;

10 если указатель текущей вершины соответствует узлу F, вернуть plan, иначе перейти к пункту 3;

Этот метод решения задачи о потоках называется по имени его создателя алгоритмом Форда-Фулкерсона. Приведенный алгоритм может быть реализован одним из двух классов программных схем: операторной (процедурной) и рекурсивной.

Решение данной задачи может быть красиво выполнено на основе рекуррентного алгоритма. В этом случае рекурсивная функция вызывается из головной функции многократно, возвращая за каждый вызов очередное улучшение плана.

Для того чтобы матрица пропускной способности Weight характеризовала направленность дуг, принято для задания прямого направления дуги использовать целое число - пропускную способность, а обратного направления - максимально возможное целое беззнаковое 0xffff.

Программа, рассчитывающая оптимальный план, может осуществлять вывод результатов в виде таблицы. Так, для графа, представленного на рис.7.1,а, результаты работы показаны в табл.7.1 и на рис.7.4.

Таблица 7.1

Начальная вершина дуги	Конечная вершина дуги	Пропускная способность	Поток
1	2	13	13
1	3	8	8
1	4	13	9
2	5	7	7
2	6	8	6
3	5	4	2
3	6	5	3
3	7	3	3
4	6	8	2
4	7	7	7
5	8	9	9
6	9	11	11
7	10	13	10
8	9	5	3
8	11	6	6
9	11	19	19
10	9	6	5
10	11	9	5
План			30

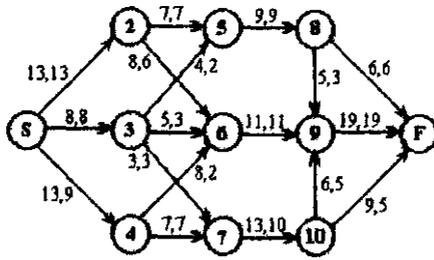


Рис. 7.4. Решение задачи распределения ресурсов

7.4. Порядок выполнения работы

1. Изучить вопросы, связанные с определением и структурой графа.
2. Досконально разобраться в алгоритме оптимизации функции потока.
3. Разработать и на любом языке программирования написать программу, рассчитывающую оптимальный поток на графе.

Индивидуальные задания:

1. Найти оптимальный поток, проходящий через граф, изображенный на рис.7.5.

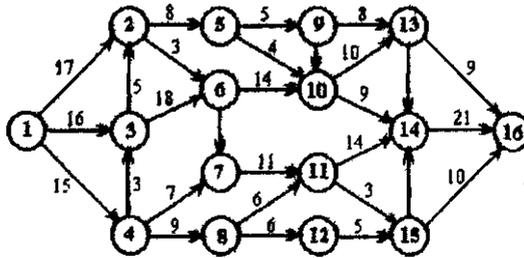


Рис.7.5

2. Найти оптимальный поток, проходящий через граф, изображенный на рис.7.6.

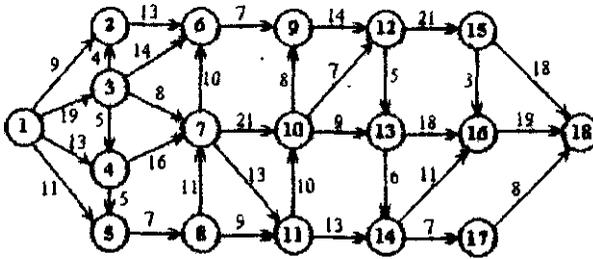


Рис.7.6

7.5. Содержание отчета

1. Цель работы.
2. Условие решаемой задачи
3. Последовательное улучшение потока, по результатам работы программы.
4. Выводы по работе, содержащие конкретные результаты решаемой задачи (таблица и изображение графа с пропускной способностью и потоком).

7.6. Контрольные вопросы

1. Что представляет собой взвешенный граф?
2. Какой поток называется оптимальным?
3. В чем суть алгоритма Форда-Фулкерсона?

ЛАБОРАТОРНАЯ РАБОТА № 8

ПОСТРОЕНИЕ, МОДЕЛИРОВАНИЕ И АНАЛИЗ ДИСКРЕТНЫХ ПРОИЗВОДСТВЕННЫХ СИСТЕМ

8.1. Цель работы

1. Ознакомление с методами имитационного моделирования, применяемыми для моделирования дискретных производственных процессов.
2. Рассмотрение ассоциативного инструмента Моделирования - сетей Петри.
3. Получение навыков формализации процессов, протекающих в производственных системах на основе механизма классических сетей Петри.

8.2. Общие сведения

8.2.1. Дискретные производственные процессы

Операция, реализуемая в технологическом объекте - это развернутое во времени целенаправленное действие, которое характеризуется своей целью и способом достижения этой цели. Операция завершается либо по достижении поставленной цели (доведение параметров объекта до заданных значений, перемещение рабочих органов в требуемые положения и т.д.), либо по истечении отведенного для нее времени, либо под влиянием внешних событий, прерывающих операцию. Множество различных операций, реализуемых в технологическом объекте, конечно. При этом операции могут повторяться неограниченное число раз.

На множестве операций вводятся *бинарные отношения следования и параллелизма*, отражающие временные связи между операциями. Две операции находятся в отношении следования, если они никогда не выполняются одновременно, т.е. момент начала любой из них сдвинут по оси времени относительно момента конца другой.

Частным случаем отношения следования является отношение непосредственного следования, когда начало одной операции совпадает с концом другой. Если две операции не находятся в отношении следования, т.е. могут выполняться одновременно (на разном оборудовании технологического объекта), то они находятся в отношении параллелизма.

Таким образом, если две операции находятся в отношении непосредственного следования, то они находятся в отношении следования (обратное в общем случае не верно); если две операции не находятся в отношении следования, то они находятся в отношении параллелизма.

Дискретным процессом, реализуемым в технологическом объекте,

называется конечное множество операций, на котором заданы бинарные отношения следования и параллелизма. Дискретный процесс всегда имеет начало и конец, и соответственно в нем фиксируется подмножество начальных и подмножество финальных операций. *Дискретный процесс* называется *циклическим*, если любая начальная операция в нем непосредственно следует за финальными и, кроме того, выполняется условие: если две операции находятся в отношении параллелизма, то они принадлежат таким частям процесса (подпроцессам), которые одновременно начинаются и заканчиваются. Если же в процессе с отмеченными выше свойствами начальные операции могут инициироваться до завершения процесса, т.е. непосредственно следовать за нефинальными операциями, то *дискретный процесс* называется *конвейерным*.

Существует два условия, которым должен удовлетворять дискретный технологический процесс, как циклический, так и конвейерный.

1. Любая операция может быть повторно начата только после своего завершения, т.е. исключается инициирование операции во время ее выполнения.

2. Дискретный процесс, начавшись, всегда может быть доведен до конца: в ходе его выполнения не должно возникать «тупиковых» ситуаций, не имеющих продолжения

Процесс считается корректным, если, во-первых, он соответствует приведенным выше двум условиям, и, во-вторых, согласован с возможностями того технологического объекта, в котором протекает.

Центральными проблемами теории дискретных технологических процессов являются задачи:

- 1) *анализа*, связанные с установлением корректности заданного процесса;
- 2) *синтеза* системы управления заданным процессом.

Решение задач анализа и синтеза основывается на использовании формальных описаний (Математических моделей) объекта и реализуемого в нем процесса, отражающих соответственно все возможные варианты поведения и конкретное поведение объекта. При анализе проверяются корректность и согласованность этих моделей, при синтезе модель процесса используется как задание, на основе которого устанавливается структура управляющего логического устройства, выбираются соответствующие средства и проводится реализация управляющего логического устройства.

В данной работе рассматриваются вопросы формализации и анализа дискретных процессов.

В настоящее время известно несколько десятков разновидностей математических схем описания дискретных процессов, наиболее часто применяемыми из которых являются сети Петри и их расширения.

Моделирующие возможности сетей Петри и их эффективность в приложениях объясняются прежде всего тем, что сеть Петри - это интеграция графа и дискретной динамической системы, она может служить, таким образом, и статической и динамической моделью представляемого с его помощью объекта. Особенную роль сети Петри играют при моделировании параллельных процессов.

здесь это едва ли не самый удобный и многообещающий инструмент исследований. Немаловажное значение имеет известное преимущество этих сетей — удобство программирования их на ЭВМ.

8.2.2. Определение сети Петри

Сети Петри - инструмент исследования систем и применяются сети Петри исключительно в моделировании. Сети Петри разрабатывались специально для моделирования тех систем, которые содержат взаимодействующие параллельные компоненты. Теория сетей Петри делает возможным моделирование системы математическим представлением ее в виде сети Петри.

В основном сети Петри рассматриваются как вспомогательный инструмент анализа, а для построения системы используются общепринятые методы проектирования. Построенная система моделируется сетью Петри, и модель анализируется. Любые трудности, встречающиеся при анализе, указывают на изъяны в проекте и для их исправления необходимо модифицировать проект. Модифицированный проект затем снова моделируется и анализируется. Этот цикл повторяется до тех пор, пока проводимый анализ не приведет к успеху [22]. Такой подход иллюстрируется на рис.8.1.

Сеть Петри состоит из четырех элементов: множество *позиций* P , множество *переходов* T , входная функция I и выходная функция O . *Входная функция* I отображает переход t_j в множество позиций $I(t_j)$, называемых входными позициями перехода. *Выходная функция* O отображает переход t_j в множество позиций $O(t_j)$, называемых выходными позициями перехода.



Рис.8.1. Сеть Петри для моделирования и анализа систем

Для иллюстрации понятий теории сетей Петри удобно работать с их (сетей Петри) графическим представлением. Структура сети Петри определяется ее позициями, переходами, входной и выходной функциями. В соответствии с этим *граф* сети Петри обладает двумя типами узлов: кружок является позицией, а планка - переходом.

Ориентированные дуги соединяют позиции и переходы, при этом некоторые дуги направлены от позиций к переходам, а другие - от переходов к позициям.

Граф сети Петри называется *ориентированным*, так как его дуги являются направленными. Вершины графа можно разделить на два множества (позиции и

переходы) таким образом, что каждая дуга будет направлена от элемента одного множества к элементу другого множества (от позиции к переходу или от перехода к позиции). Следовательно, граф сети Петри можно также назвать *двудольным ориентированным графом* (рис.8.2).

Сеть Петри определяется не только структурой, но и маркировкой. *Маркировка* M - это распределение фишек по позициям сети. *Фишка* - примитивное понятие сети Петри, которое используется для определения выполнения сети. Положение фишек при выполнении сети Петри может меняться, что приводит к изменению маркировки.

Сеть Петри выполняется посредством запуска переходов. Переход *запускается (активируется)* удалением фишек из его входных позиций. Переход выполняется в течение заданного времени, после чего наступает его *пассивизация*. Момент пассивизации перехода связан с образованием новых фишек, помещаемых в его выходные позиции.

Переход может быть запущен только тогда, когда все его входные позиции имеют фишки; в этом случае говорят, что *переход разрешен*. Активизация перехода в целом заменяет маркировку M сети Петри на новую маркировку $M1$ ($|M| > |M1|$). По истечении времени выполнения перехода, он пассивизируется, при этом маркировка $M1$ сети Петри заменяется на маркировку $M2$ ($|M1| < |M2|$).

Простое представление системы сетью Петри основано на двух понятиях: событиях и условиях [22]. *Событие* - это действие, имеющее место в системе. Возникновением событий управляет состояние системы, которое в свою очередь может быть описано множеством условий. *Условие* - это логическое описание состояния системы, оно может принимать значение "истина" либо значение "ложь".

Так как события являются действиями, то они могут происходить. Для того чтобы событие произошло, необходимо выполнение соответствующих условий. Эти условия называются *предусловиями* события. После того как событие произошло, выполняются иные условия - *постусловия*.

В качестве примера рассмотрим задачу моделирования простого автомата-продавца. Автомат-продавец находится в состоянии ожидания до тех пор, пока не Появится заказ, который он выполняет и посылает на доставку. Условиями для такой системы являются: автомат-продавец свободен и ждет (Saw); заказ прибыл и ждет (Spw); заказ выполнен (Spm). Событиями будут: заказ поступает (Trc); автомат-продавец выполняет заказ (Tam); заказ посылается на доставку (Trt).

События, условия и постусловия модели системы автомата-продавца приведены в табл.8.1.

Таблица 8.1

Событие	Предусловие	Постусловие
Trc	Нет	Spw
Tam	Spw, Saw	Spm, Saw
Trt	Spm	Нет

Такое представление системы легко моделировать сетью Петри. В сети Петри условия моделируются позициями, события - переходами, при этом входы перехода являются предусловиями соответствующего события; выходы - постусловиями. Возникновение события равносильно запуску соответствующего перехода. Выполнение условия представляется фишкой в позиции, соответствующей этому условию. Активизация перехода удаляет разрешающие фишки, представляющие выполнение предусловий; пассивизация перехода образует новые фишки, которые представляют выполнение постусловий [22].

Сеть Петри на рис.8.2 иллюстрирует Модель описанного выше автомата-продавца.

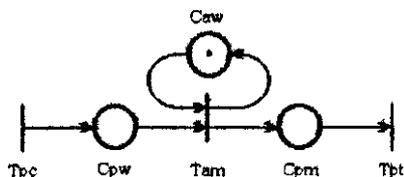


Рис.8.2. Сеть Петри, моделирующая работу автомата-продавца

8.3. Интерпретатор сетевых моделей

8.3.1. Алгоритм функционирования сети Петри

Для обработки моделей систем, выполненных на основе сетей Петри, необходимо создать программу-интерпретатор, способную воспроизводить динамику процессов на сети Петри. Ключевой идеей такого интерпретатора будет обработка последствий свершения особых событий на сети. Особое событие напрямую определяется моментом пассивизации какого-либо перехода.

Обработка последствий особого события потребует наличия следующих процедур:

passiveTransition (сокращенно **pT**) - моделирование действий перехода при передаче меток в его выходные позиции. Передачу меток необходимо выполнить до рассмотрения других фаз процесса на сети Петри с целью отображения одновременности изменения разметки в выходных позициях перехода;

pastPassive (pP) - организация обработки последствий изменения разметки в выходных позициях пассивизируемого перехода. Последовательность просмотра таких позиций определяет приоритет очередных событий на сети Петри;

possibleActive (pA) - проверка возможности активизации переходов, находящихся на выходных дугах позиций, с только что изменившейся разметкой;

activeTransition (aT) - активизация перехода, если необходимость в этом установлена процедурой possibleActive. Активизация перехода требует изменения разметки в его входных позициях и планирования в момент $time+d(t)$ ($time$ - текущее модельное время, $d(t)$ - время выполнения перехода) пассивизации

перехода t , т.е. нового особого события;

fixed Event (fE) - фиксация особого события на сети Петри в списке событий;

directEvent (dE) - выбор очередного особого события на сети и начало нового шага моделирования, если список событий не пуст;

После определения необходимых процедур можно записать алгоритм моделирования процессов на сети Петри а рекуррентном виде, указывая вложенность перечисленных процедур:

$$\text{Event}(t+1) = \text{dE}(\text{pT}(\text{pP}(\text{pA}(\text{aT}(\text{fE}(\text{Event}(t))))))),$$

где $\text{Event}(t)$ - особое событие, соответствующее переходу t .

Рассматриваемая рекуррентная структура алгоритма отображает одну из эффективных, но не единственных схем его построения.

Прежде чем продолжить обсуждение алгоритма, остановимся на некоторых основных понятиях теории моделирования.

8.3.2.Список запланированных событий

Для оптимальной реализации концепции переменного приращения все планируемые особые события помещаются в динамически изменяемый линейный список со связанным распределением. При планировании особого события вычисляется время его наступления, в список добавляется новый элемент, элемент располагается в списке в соответствии со значением времени Наступления события (чем меньше это время, тем ближе к началу списка расположен элемент). Для выбора очередного запланированного особого события достаточно лишь обратиться к первому элементу списка, и после всех обработок удалить его.

Список со связанным распределением организуется в любом языке высокого уровня с помощью структурированного типа данных, где наряду с информационными полями находится указатель на смежные элементы списка [23].

8.3.3.Описание структуры сети Петри

Так как граф сети Петри представляет собой двудольный ориентированный граф, для его фиксации достаточно определить два массива. Первый из них хранит списки номеров входных позиций перехода, второй - списки номеров выходных позиций перехода. Описание сети Петри завершается определением списков временных задержек в выполнении переходов и начальной разметки позиций сети. В связи с этим, структура используемой сети Петри для работы Интерпретатора фиксируется в виде .INI-файла (Petri.ini) со следующими разделами: Transition - список имен переходов и их временных задержек, Condition - список имен позиций и их начальной разметки, Transition-Condition - список связности переходов с позициями, Condition-Transition - список выходных переходов для позиций, ModelTime - общее время моделирования, FirstTransition - имя перехода, который должен выполняться первым. Формат файла представлен на рис.8.3.

Имя перехода: временная задержка | начальное значение - конечное значение;

Имя перехода: временная задержка | начальное значение - конечное значение;

Condition:

Имя позиции;

Имяпозиции;

Transition-Condition:

Имя перехода [, Rand]: имя позиции 1, имя позиции 2,...;

Имя перехода [, Rand]: имя позиции 1, имя позиции 2,...;

Condition- Transition:

Имя позиции [, Rand]- имя перехода 1, имя перехода2,...;

Имя позиции [, Rand]: имя перехода 1, имя перехода2,...;

ModelTime:

Требуемое время моделирования:

FirstTransition:

Имя первого перехода модели.

Рис.8.3. Формат файла Petri.ini

Ключевое слово Rand сигнализирует о том, что фишки после пассивизации перехода помещаются в выходные позиции в произвольном порядке (порядок влияет на порядок запуска выходных переходов данных позиций). Если ключевое слово Rand не указано, то порядок изменения разметки в позициях будет соответствовать порядку их перечисления. Файл инициализации должен помещаться в текущем каталоге и иметь имя Petri.ini. Однако можно использовать и другие имена. В этом случае интерпретатор Запускается с параметром - именем файла инициализации.

8.4. Примеры построения моделей на основе сетей Петри

Введение параллелизма полезно только в том случае, когда компоненты процессов могут взаимодействовать при получении решения задачи. Такое взаимодействие требует распределения ресурсов между процессами [21]. Для гарантии правильности работы системы в целом распределением необходимо управлять. Проблемы синхронизации, возникающие при взаимодействии процессов, иллюстрируются многочисленными примерами. Среди задач синхронизации: задача о взаимном исключении, производителе/потребителе, обедающих мудрецах, чтении/записи и др. Эти задачи стали классическими в области синхронизации; каждое новое предложение по механизму синхронизации должно решать их. И хотя сети Петри представляют собой схему моделирования, а не механизм синхронизации, они определенно способны моделировать процессы синхронизации.

Рассмотрим использование сети Петри для решения задач: об обедающих мудрецах и курильщиках сигарет.

8.4.1 .Задача об обедающих мудрецах

Задача об обедающих мудрецах была предложена Дейкстрой [22]. Заключается она в следующем. Пять мудрецов сидят за круглым столом, на котором находятся блюда китайской кухни. Между соседями лежит одна палочка для еды. Однако для приема пищи необходимо две палочки, следовательно, каждый мудрец должен взять палочку слева и палочку справа от себя. Проблема, конечно, заключается в том, что если все мудрецы возьмут палочки слева и затем будут ждать, когда освободятся палочки с правой стороны, то они будут ждать вечно, т.е. возникнет состояние тупика. На рис.8.4 показано решение этой задачи с помощью сети Петри. Позиции $C_1...C_5$ представляют палочки для еды. Каждый мудрец представлен одной позицией M_i и одним переходом E_i для состояния размышления и принятия пищи соответственно. Для того чтобы активизировался переход E_i , т.е. мудрец перешел в состояние принятия пищи, обе палочки (слева и справа) должны быть свободны.

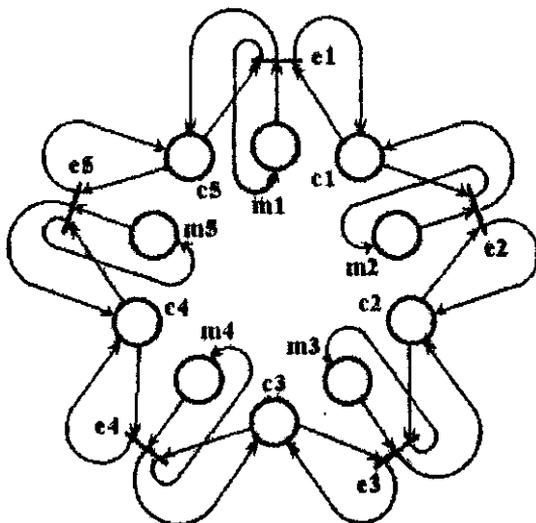


Рис.8.4. Решение задачи о мудрецах с помощью сети Петри

События, условия и постусловия модели задачи об обедающих мудрецах приведены в табл.8.2.

Таблица 8.2

Событие	Предусловие	Постусловие
E_i	C_{i-1}, C_i, M_i	C_{i-1}, C_i, M_i

Любая сеть Петри представима в *стандартной форме*. Введение стандартной формы позволяет определить процесс функционирования сети относительно ее начальной и конечной разметок. Стандартная форма сети Петри определяется вводом, так называемых, стартового (Ts) и финишного (Tf) переходов. Введение переходов Ts и Tf означает, что процесс на сети Петри начинается в нулевой момент времени активизацией перехода Ts и завершается пассивацией перехода Tf. Переход Ts в общем случае формирует начальную разметку исходной сети Петри, активизация перехода Tf возвращает сеть в исходное состояние.

Для представления построенной ранее сети Петри в стандартной форме вводится переход Ts с временной задержкой 0 единиц. Выходными дугами данный переход соединен с позициями C1...C5 и M1...M5.

Когда речь идет о функционировании сети Петри в течение какого-либо времени Tend, то целесообразно в качестве финишного перехода вводить фиктивный переход Tf, активизируемый в начальный момент времени и имеющий задержку, равную Tend. Моделирование, таким образом, продолжается до тех пор, пока не наступит пассивация перехода Tf.

В процессе функционирования сети Петри могут возникать особые ситуации, когда Некоторая позиция порождает взаимоисключающий альтернативный процесс активизации нескольких переходов. Для рассматриваемой задачи, например, позиция C1 с разметкой может запустить как переход E1, так и переход E2 (при наличии меток в остальных входных позициях перечисленных переходов). В таких ситуациях назначается приоритет активизации конкурирующих переходов.

На основании описанного выше условия задачи и построенной сети Петри формируется файл инициализации модели (рис.8.5). Интерпретатор должен моделировать поведение обедающих мудрецов в течение 480 единиц времени (имеется в виду минут, т.е. продолжительность рабочего дня). Считается, что на еду среднестатистическому мудрецу требуется 30 ± 15 минут.

В процессе работы программы на экран выводится информация о текущем состоянии системы, как то: текущее модельное время, общее время работы всех переходов, имя активизируемого перехода и запланированное время его пассивации. После работы программы ко всем перечисленным данным добавляется еще и коэффициент работы для каждого перехода (коэффициент занятости мудреца). Этот коэффициент вычисляется как отношение времени, в

Таблица 8.3

Переход	Время работы	Коэффициент
E1	184	38%
E2	189	39%
E3	211	44%
E4	188	39%
E5	188	39%

течение которого переход был запущен к общему времени моделирования, выраженное в процентах. Оценивая данные, полученные в результате работы программы (результаты одного из запусков программы показаны в табл.8.3), можно констатировать, что рассматриваемая модель синхронизирует процессы таким образом, что каждому мудрецу отводится, по возможности, одинаковое i -ремя на принятие пищи.

```
Transition.  
T0: 0;  
E1- 15-45;  
E2. 15-45;  
E3: 15-45;  
E4: 15-45;  
E5: 15-45;  
Condition:  
M1;  
M2;  
M3;  
M4;  
M5;  
C1;  
C2;  
C3;  
C4;  
C5;  
Transition-Condition:  
T0, Rand: M1, C1, M2, C2, M3, C3, M4, C4, M5, C5;  
E1, Rand: M1, C1, C5;  
E2, Rand: M2, C1, C2;  
E3, Rand: M3, C2, C3;  
E4, Rand: M4, C3, C4;  
E5, Rand: M5, C4, C5;  
Condition-Transition:  
M1: E1;  
M2: E2;  
M3: E3;  
M4: E4;  
M5: E5;  
C1, Rand: E1, E2;  
C2, Rand: E2, E3;  
C3, Rand: E3, E4;  
C4, Rand: E4, E5;  
C5, Rand: E1, E5;  
ModelTime:  
480;  
FirstTransition:  
T0
```

Рис.8.5. Файл инициализации модели задачи об обедающих мудрецах

8.4.2. Задача о курильщиках сигарет

При непосредственном решении большинства задач синхронизации сетями Петри возникают трудности, которые могут быть разрешены на основе известных механизмов синхронизации. В частности, одним из самых популярных механизмов синхронизации являются P- и V-операции над семафорами, впервые определенные Дейкстрой. *Семафор* - это элемент данных, который может принимать только неотрицательные целые значения. V-операция увеличивает значение на 1, а P-операция уменьшает его на 1. P-операцию можно применять только в том случае, когда значение семафора останется в результате неотрицательным; если же значение семафора равно 0, то P-операция должна ждать, пока какой-нибудь другой процесс не выполнит V-операцию. P- и V-операции определены как примитивные, т.е. никакая другая операция не может изменять значение семафора одновременно с ними.

Такие операции легко моделируются сетью Петри, как показано на рис.8.6. Каждый семафор моделируется позицией, количество фишек в позиции показывает значение семафора. P-операция использует позицию семафора в качестве входа, V-операция - в качестве выхода. Таким образом, сети Петри позволяют моделировать все те системы, которые проектируются и описываются с помощью семафоров [22].

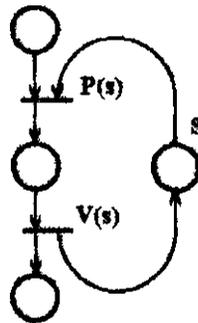


Рис.8.6. P- и V-операции

Однако сети Петри справляются и с теми задачами, которые не могут быть успешно решены с помощью семафоров. Патил в 1971г. предложил доказательство того, что P- и V-операции - недостаточно мощное средство для решения всех задач координации. Он сформулировал задачу синхронизации, которая не может быть решена с помощью P- и V-операций, - это задача о курильщиках сигарет.

Задача о курильщиках сигарет включает (по меньшей мере) четыре процесса, которые моделируют агента и трех курильщиков. Каждый курильщик непрерывно изготавливает сигарету и курит ее. Для этого необходимы три составляющие части: табак, бумага и спички. Один из курильщиков всегда имеет бумагу, другой - табак, а третий - спички. Агент обладает бесконечными запасами всех трех составных частей. Агент кладет две составные части на стол.

Курильщик, имеющий третий недостающий ингредиент, может сделать и закурить сигарету, сигнализируя об этом агенту. Тогда агент помешает другие два из трех ингредиентов, и цикл повторяется.

Если семафор поставить в соответствие каждой составной части, задача о курильщиках формулируется в терминах семафоров. Семафоры первоначально равны нулю. Агент увеличивает два из трех семафоров с помощью V-операций, а затем ждет семафора «сделано». Соответствующий процесс курильщика уменьшает два семафора (с помощью P-операций), затем, произведя действие «сделать сигарету» и «закурить сигарету», увеличивает семафор, указывая «сделано». Задача заключается в том, чтобы разработать программу процессов курильщиков для определения какой из трех процессов должен действовать в очередной момент. Действия агента фиксированы и не могут быть изменены.

Патил доказал, что никакая последовательность P- и V-операций не может корректно решить эту задачу. Это было показано с помощью доказательства того, что все P- и V- «решения» могут быть промоделированы сетями Петри определенного вида - каждый переход имеет не более двух входов.

Однако в данном случае решением является сеть Петри другого вида, хотя и нет способа преобразовать сеть Петри одного вида в сеть другого вида, не допуская возможности возникновения тупика

На рис.8.7 показана сеть Петри, решающая задачу о курильщиках сигарет.

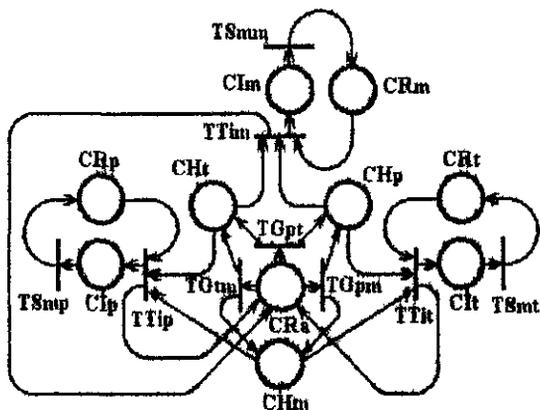


Рис.8.7. Решение задачи о курильщиках сигарет

Окончания в названиях переходов и позиций соответствуют: m - курильщику со спичками, t - курильщику с табаком, p - курильщику с бумагой. В модели присутствуют следующие позиции: CR - курильщик готов и ожидает, CI - готовность к курению, CH - в наличии имеется ингредиент (т - спички, t - табак, p - бумага), Cra - готовность агента к выделению ингредиентов. Рассматриваемая сеть Петри содержит следующие переходы: TSm - процесс курения, Tti - получение курильщиком ингредиентов, TO - агент выкладывает ингредиенты (pt

- бумагу и табак, pm - бумагу и спички, tm - табак и спички).

События, условия и постусловия модели задачи о курильщиках, с учетом стартового перехода, приведены в табл.8.4.

Таблица 8.4

Событие	Предусловие	Постусловие
T0		CRm, CRt, CRp, CRa
TGpt	CRa	CHt, CHp
TGpm	CRa	CHm, CHp
TGtm	CRa	CHm, CHt
TTim	CHt, CHp, CRm	Clm, CRa
TTit	CHm, CHp, CRt	Clt, CRa
TTip	CHt, CHm, CRp	Clp, CRa
TSmm	Clm	CRm
TSmt	Clt	CRt
tSmp	Clp	CRp

В табл.8.5 приведены результаты работы программы после одного из запусков (при установленном времени моделирования 180 единиц времени).

Таблица 8.5

Переход	Время работы	Коэффициент
TSmm	130	72%
TSmt	140	78%
TSmp	124	69%

8.5. Порядок выполнения работы

1. Изучить вопросы, связанные с имитационным моделированием дискретных процессов.

2. Досконально разобраться в алгоритме функционирования интерпретатора сетевых моделей и в методике построения сетей Петри.

3. В соответствии с выбранным вариантом индивидуального задания определить условия и события модели, описать переходы и позиции.

4. Подготовить файл инициализации модели и выполнить моделирование описанной системы.

5. Оценить результаты моделирования и при необходимости подкорректировать модель.

Индивидуальные задания.

1. Из литейного цеха на участок обработки и сборки поступают заготовки через 20 ± 5 мин. Треть из них обрабатывается в течение 60 минут и поступает на комплектацию. Две трети заготовок обрабатывается за 30 минут перед комплектацией, которая требует наличия одной детали первого типа и двух деталей второго. После этого все три детали подаются на сборку, которая занимает 60 ± 2 мин. для первой детали и 60 ± 8 мин. для двух других, причем они участвуют в сборке одновременно. При наличии на выходе одновременно всех трех деталей изделие покидает участок. Смоделировать работу участка в течение 100ч.

2. В машинный зал с интервалом времени 10 ± 5 мин. заходят пользователи, желающие произвести расчеты на ЭВМ. В зале имеется одна ЭВМ, работающая в однопрограммном режиме. Время, необходимое для решения задач, включая вывод результатов на печать, характеризуется интервалом 15 ± 5 мин. Третья часть пользователей после окончания решения своей задачи производит передачу результатов работы программы по локальной сети (продолжительность передачи - 3 ± 2 минут). Смоделировать процесс обслуживания 100 пользователей.

8.6. Содержание отчета

1. Цель работы.
2. Условие решаемой задачи.
3. Определение событий и условий в соответствии с выбранной задачей.
4. Характеристики рассматриваемой системы по результатам проведенных модельных экспериментов.
5. Выводы по работе, содержащие конкретные результаты решаемых задач.

8.7. Контрольные вопросы

1. Что такое технологические операции?
2. Какие процессы называются дискретными и чем характеризуется их корректность?
3. Перечислить основные проблемы, рассматриваемые в теории дискретных технологических процессов.
4. Определение сети Петри.
5. В чем заключается алгоритм функционирования интерпретатора сетей Петри?
6. Что такое семафоры?

7.ЛИТЕРАТУРА(НЕПРАВЛЕНО!!!)

1. Управление машиностроительным предприятием: Учебник для машиностр. спец. вузов/ С.В.Смирнов, С.Н.Ефимушкин, А.А.Колобов и др.; Под ред. С.Г.Пуртова, С.В.Смирнова.-М.:Высш.шк.,1989.
2. Белман Р. Динамическое программирование: Пер. с англ./ Под ред. Н.Н.Воробьева.- М: ИЛ, 1960.
3. Белман Р., Дрейфус С. Прикладные задачи динамического программирования: Пер. с англ./ Под ред. А.А.Первозванского.-М.: Наука,1965.
4. Вентцель Е.С.Исследование операций.-М.:Сов.радио, 1972.
5. Грешилов А.А. Как принять наилучшее решение в реальных условиях.-М.:Радио и связь,1991.
6. Системное проектирование взаимодействия человека с техническими средствами.- М.:Высш.шк.) 1991.
7. Мамиконов А.Г. Основы построения АСУ.-М.:Высш.шк., 1981.
8. Общесистемное проектирование АСУ реального времени/Под ред.В.А.Шебалина.-М.:Радио и связь,1984.
9. Мельников Ю.Н. Достоверность информации в сложных системах.- М.: Радио и связь,1985.
10. Рахимов Т.Н., Заикин О.А., Советов В.Я. Основы построения АСУ. Ташкент: Укитувчи, 1984.
11. Мельцер МИ. Разработка алгоритмов АСУП. М., 1975.
12. Первозванский А.А. Математические модели в управлении производством. М.,1975.
13. Планирование дискретного производства в условиях АСУ/ Под ред. В.М.Глушкова. Киев, 1975.
14. Ильичев А.В., Волков В.Д., Грушанский В.А. . Эффективность проектируемых элементов сложных систем. - М.: Высш.шк., 1982.
15. Денисов А.А., Колесников Д.Н. Теория больших систем управления. - Л.: Энергоиздат, 1982.
16. Стеганцев М.А. Вероятность проводки самолета системой наземных РЛС. Теория и техника радиолокации, радионавигации и радиосвязи в гражданской авиации. - Рига, 1997.
17. Советов Б.Я., Яковлев С.А. Моделирование систем: Учебник для вузов по спец. «Автоматизированные системы управления». - М.: Высш.шк.,

1985.-217с.

18. Шрайбер Т.Дж. Моделирование на OPSS: Пер. с англ. - М.: Машиностроение, 1980. - 592с.

19. Лэнгсам И., Огенстайн М., Тененбаум А., Структуры данных для персональных ЭВМ: Пер. с англ. -М.: Мир, 1989. -568с.

20. Свами М., Тхуласираман К., Графы, сети и алгоритмы: Пер. с англ. - М.:Мир, 1984-455с.

21. Юдицкий С.А., Магергут В.З. Логическое управление дискретными процессами. Модели, анализ, синтез: - М.: Машиностроение, 1987. - 176с,

22. Питерсон Дж. Теория сетей Петри и моделирование систем. - М.: Мир, 1984,-264с.

23. Кнут Д. Искусство программирования для ЭВМ. Т.1. Основные алгоритмы: - М.: Мир, 1976. - 734с.