

621,7
К49

2077

Министерство образования Республики Беларусь
**БЕЛОРУССКАЯ ГОСУДАРСТВЕННАЯ ПОЛИТЕХНИЧЕСКАЯ
АКАДЕМИЯ**

Кафедра «Технология машиностроения»

**Ф. Ф. Климович
А. Ф. Присевко**

МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ

ТЕХНОЛОГИЧЕСКИХ ЗАДАЧ В МАШИНОСТРОЕНИИ

Учебно-методическое пособие по лабораторным работам
для студентов машиностроительных специальностей высших
учебных заведений

Минск 2000

Министерство образования Республики Беларусь
БЕЛОРУССКАЯ ГОСУДАРСТВЕННАЯ ПОЛИТЕХНИЧЕСКАЯ
АКАДЕМИЯ

Кафедра «Технология машиностроения»

Ф.Ф.Климович
А.Ф.Присевко

**МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ
ТЕХНОЛОГИЧЕСКИХ ЗАДАЧ В МАШИНОСТРОЕНИИ**

**Учебно-методическое пособие по лабораторным работам
для студентов машиностроительных специальностей
высших учебных заведений**

Рекомендовано Министерством образования Республики Беларусь
в качестве учебно-методического пособия
для студентов специальности Т.03.01 –
«Технология, оборудование и автоматизация машиностроения»

Минск 2000

Климович Ф.Ф., Присевок А.Ф. Математическое моделирование технологических задач в машиностроении. Учебно-методическое пособие по лабораторным работам для студентов машиностроительных специальностей высших учебных заведений. – Мн.: ВГПА, 2000. - 88 с.

В издании приведены лабораторные работы по дисциплине “Математическое моделирование технологических задач в машиностроении”, выполнение которых предполагает разработку и использование математических моделей и алгоритмов, нашедших наиболее широкое использование при решении трудоёмких проектно-технологических задач с целью автоматизации и поиска оптимальных решений при проектировании и разработке технологических процессов. В работе приведены также понятия и алгоритмы математических дисциплин, необходимые для выполнения работ, а также методологические аспекты математического моделирования. Лабораторный практикум предназначен для студентов дневной и заочной формы обучения машино- и приборостроительных специальностей высших учебных заведений.

Рецензенты:

В.В. Базылько; А.Н. Рязанцев

Введение

Важную роль в подготовке инженеров технологических специальностей для различных отраслей машино- и приборостроения наряду с изучением и проработкой основных курсов базовых дисциплин, таких как «Технология машиностроительных материалов», «Технология машиностроения», «Материаловедение» и др., прохождением учебных и конструкторско-технологических практик, выполнением курсовых и дипломного проектов играет изучение математических дисциплин. Дисциплина «Математическое моделирование технологических задач в машиностроении» занимает среди них особое место, она является связующим звеном между известными положениями, методами и алгоритмами различных математических дисциплин и прикладными задачами технологического и конструкторского проектирования. Она призвана обучить студентов приемам и навыкам использования полученных ими математических знаний для более качественного решения традиционных и новых технологических задач.

Настоящее учебное пособие предназначено для студентов машиностроительных и приборостроительных специальностей высших учебных заведений и в основном соответствует программе курса «Математическое моделирование технологических задач в машиностроении». Данная дисциплина основывается на знании технологии машиностроения и таких математических дисциплин, как теория множеств, теория графов, математический анализ, алгебра, линейное и динамическое программирование. Знания и навыки, приобретенные студентами при изучении указанной дисциплины, могут быть использованы при решении различных практических задач технологии машиностроения. С другой стороны, эти знания необходимы при изучении многих других дисциплин. В первую очередь это касается таких дисциплин, как «Основы САПР» и «САПР технологических процессов», так как основой различных систем автоматизированного проектирования является их математическое обеспечение, содержащее математические модели различных проектных задач.

Широкое применение для решения задач технологии машиностроения находят методы дискретной математики и, в частности, теории графов. Это обусловлено тем, что многие объекты машиностроительного производства обладают ярко выраженной дискретностью. Типичными примерами дискретного являются штамповочное, сборочное или механообрабатывающее производство, где технологический процесс представляет собой множество технологических операций обработки на ряде станков. Предприятие, представляющее собой совокупность цехов и служб, сами цеха, участки также являются дискретными объектами. Это обуславливает актуальность дискретной математики в системе подготовки инженеров-технологов машино- и приборостроительного профиля.

При решении технологических задач методом математического моделирования наиболее часто применяются алгоритмы поиска оптимальных путей в

графах. Эти алгоритмы можно использовать в тех случаях, когда существует конечное число вариантов решения прикладной задачи, от выбора которых зависит значение некоторого критерия оптимальности, например, себестоимости, времени обработки, энергопотребления и т.п. В частности, такой подход может быть использован для выбора оптимальной структуры перехода, операции, а также для оптимизации технологического маршрута в целом, когда обработка детали допускает различные его варианты.

Выполнение лабораторных работ №1–6 позволит приобрести навыки использования методов дискретной математики для решения различных задач технологии машиностроения.

Лабораторная работа №7 посвящена использованию метода линейного программирования для решения прикладных технологических задач. Без преувеличения можно сказать, что среди математических методов последний получил наиболее широкое применение при решении различных прикладных задач. Классическим примером его использования в технологии машиностроения является задача выбора оптимальных режимов резания.

Лабораторная работа №8 носит научно-исследовательский характер. Здесь студентам предлагается разработать имитационную математическую модель формообразования поверхностей деталей при их обработке на металлорежущих станках с силовым замыканием инструмента и заготовки. При этом устанавливается аналитическая, количественная связь между параметрами точности обработки, с одной стороны, и параметрами, характеризующими состояние технологической системы, с другой.

Решение этой задачи позволяет количественно оценить влияние на точность обработки таких параметров технологической системы «станок–приспособление–инструмент–деталь», как износ трущихся элементов кинематических пар станка, погрешности схемы обработки, погрешности установки и закрепления инструмента и заготовки, погрешности изготовления и сборки узлов станка, упругие и тепловые деформации и т.п. Разработка такой математической модели позволяет решать ряд практических задач, например, определить минимальные требования к точности изготовления конкретных узлов и механизмов проектируемой технологической машины, обеспечивающие требуемые параметры качества обработанных поверхностей с заданной производительностью.

Рассматриваемая лабораторная работа отличается также наибольшей методологической ценностью, поскольку предусматривает проработку всех этапов решения задачи методом математического моделирования от постановки цели исследования до выполнения численных исследований с помощью разработанной компьютерной программы.

В разделах 8 и 9 изложены понятия и алгоритмы дискретной математики и теории линейного программирования. Эта информация позволяет выполнить приведенные лабораторные работы без привлечения дополнительной литературы.

1. МЕТОД МАТЕМАТИЧЕСКОГО МОДЕЛИРОВАНИЯ

1.1. Сущность метода математического моделирования

Математика представляет собой средство или инструмент для решения различных задач, возникающих в других отраслях науки и в практической деятельности человека. Этим инструментом, как и любым другим, надо уметь пользоваться, т.е. уметь использовать её широкие возможности для решения других задач. Методы математической формализации и решения задач, возникающих в различных областях деятельности человека, объединяют под общим названием *математическое моделирование*. Общепринятого определения этого термина не существует. На эмоциональном уровне можно сказать, что математическое моделирование - это искусство применять математику. В философском контексте математическое моделирование является одним из наиболее общих методов научного познания закономерностей создания и функционирования реальных объектов различной природы.

При решении задач методом математического моделирования объект, подлежащий изучению (реальная технологическая система, процесс, производственная ситуация, проектная задача и т. п.), заменяется математической моделью. *Математическая модель* представляет собой совокупность математических соотношений, отображающих взаимосвязь между существенными с точки зрения решаемой задачи параметрами (признаками) объекта-оригинала. Математические соотношения могут представлять собой функциональные зависимости или логические соотношения. Под признаками моделируемого объекта понимаются параметры его структуры, различные его свойства, особенности и закономерности его функционирования и т.п.

Практически любой реальный объект обладает достаточно большим числом характеризующих его признаков. Разработка математических моделей, описывающих всевозможные взаимосвязи между всеми параметрами моделируемого объекта, в подавляющем большинстве случаев просто невозможна из-за их невероятной сложности. С другой стороны, как правило, это и не требуется, так как в зависимости от предприняемого исследования те или иные параметры и их взаимосвязи являются просто несущественными и могут быть проигнорированы. Так, при моделировании движения маятника такие его параметры, как коэффициент упругой деформации или цвет его покрытия не являются существенными.

Если математическая модель построена корректно, то существенные признаки её и объекта оригинала идентичны, со степенью достоверности, достаточ-

ной с точки зрения решаемой задачи. В соответствии с этим знания, полученные при исследовании модели, переносятся на оригинал.

В настоящее время основные этапы перехода от объекта-оригинала к его математической модели не формализованы. Эти этапы выполняются каждым исследователем на основе опыта, интуиции и анализа аналогичных исследований. При этом проблема считается разрешимой методом математического моделирования, если удастся ее свести к математическим задачам, для которых известны методы решения. В противном случае прикладная проблема сводится к математической проблеме, для разрешения которой должны быть разработаны новые математические теории.

1.2. Этапы решения задачи методом математического моделирования

Исследование реальных объектов методом математического моделирования в общем случае представляет собой последовательное выполнение следующих этапов.

1. Формулировка цели исследования, которая должна быть достигнута при моделировании. Не следует умалять важность этого этапа, так как целью исследования в значительной степени определяется сложность математической модели и сама возможность решения задачи.
2. Анализ моделируемого объекта. На этом этапе устанавливается, какие из признаков объекта оригинала являются существенными с точки зрения решаемой задачи, возможно, определяются исходные данные, характеризующие эти признаки, возможна разработка конструктивной или функциональной схемы объекта и т.п.
3. Разработка физической модели и обоснование её соответствия по существенным признакам моделируемому объекту. На этом этапе формулируются ограничения и предположения, которые упрощают описание объекта оригинала, не нанося при этом существенного ущерба достоверности получаемых результатов исследования. Этот этап важен, поскольку реальные объекты в своем большинстве крайне сложны, и без подобного упрощения их математическое моделирование зачастую просто невозможно.
4. Разработка математической модели.

5. Разработка алгоритма или алгоритмов компьютерной модели. Здесь следует отметить, что на основании одной и той же модели могут быть построены различные алгоритмы. Это зависит от того, какие из параметров в рамках принимаемого исследования будут известными, а какие искомыми. Например, параметры, которые считаются известными при решении задач анализа, будут неизвестными (искомыми) при решении задач синтеза и наоборот.
6. Разработка компьютерной программы или пакета программ. Обычно для этих целей используют языки программирования высокого уровня, такие как Паскаль, Си, Бейсик, Delphi и др.
7. Отладка и тестирование программ. Этот этап часто является самым трудоемким. Здесь выявляются ошибки не только в программе, но и в алгоритме, и в самой математической модели. Нередко на этапе отладки программ выявляется некорректность или недопустимость допущений и ограничений, принятых на этапе разработки физической модели. Это в свою очередь может привести к необходимости отказа от разработанной математической модели и разработки новой.
8. Численные исследования, анализ и интерпретация результатов численных исследований в терминологии объекта оригинала.

Приведенная схема решения задач методом математического моделирования является в определенной степени условной. Так, не всегда есть необходимость в допущениях и ограничениях, устанавливаемых на третьем этапе, решение многих задач не требует привлечения вычислительной техники, иногда возможно использование известных алгоритмов и готовых компьютерных программ и т.п.

1.3. Погрешности результатов при математическом моделировании

Необходимо помнить, что при решении определенного класса задач методом математического моделирования неизбежны погрешности получаемых результатов, которые по источнику их происхождения можно разбить на четыре группы.

1. Погрешности формализации. Возникают при переходе от объекта оригинала к его физической модели и обусловлены приближенным его соответствием этой модели. Количественную оценку этих погрешностей в общем случае

указать нельзя. Обычно такую оценку получают при экспериментальных исследованиях на самих объектах оригиналах или на их макетах.

2. Погрешности исходных данных связаны с физическими измерениями объекта оригинала. При измерениях в технических системах погрешность измерения в статических условиях обычно составляет 1 - 5%, погрешность измерения динамических величин 5 - 20%. Это приводит к погрешностям получаемых результатов.

3. Погрешности вычислительного алгоритма связаны с приближенным решением математических соотношений численными методами. Погрешность вычислительного алгоритма должна быть в 2...5 раз меньше погрешности исходных данных. Следует иметь в виду, что неудачно разработанный алгоритм может привести к тому, что результаты численного решения не будут соответствовать исследуемой математической модели.

4. Погрешности машинного округления связаны с тем, что в ЭВМ все вычисления выполняются с определенным числом значащих цифр. Эта погрешность зависит от типа компьютера и программы. Погрешность машинного округления должна быть в 5...10 раз меньше погрешности вычислительного алгоритма.

Таким образом, для получения достоверных результатов при математическом моделировании необходимо глубокое понимание всех существенных особенностей исследуемого объекта, его модели, вычислительных алгоритмов, программ, возможностей вычислительной техники.

2. ЛАБОРАТОРНЫЕ РАБОТЫ

Лабораторная работа № 1

ОПТИМИЗАЦИЯ ТЕХНОЛОГИЧЕСКИХ МАРШРУТОВ МЕХАНООБРАБОТКИ



Цель работы

Овладение навыками разработки математических моделей технологических маршрутов механообработки и их оптимизации путем использования методов теории графов.

Постановка задачи

На вход ГПС поступила партия заготовок для изготовления деталей с подобными конструктивно-технологическими признаками. Разработан технологический маршрут механической обработки заготовки, известны станки, которыми оснащена ГПС. Каждая операция k может быть выполнена на одном из нескольких взаимозаменяемых станков m , установлено время p_{km} её выполнения на каждом из них (см. таблицу в соответствии с вариантом задания). Известно также время транспортировки заготовки от станка к станку, которое следует определять из выражения

$$t_{ij} = \begin{cases} 0, & \text{если } i = j; \\ [(i + j + n) \bmod (p_{\max} - 1)] + 1, & \text{если } i \neq j, \end{cases}$$

где i, j - номера станков; t_{ij} - время транспортировки заготовки от одного из них к другому; n - номер варианта задания;

$$p_{\max} = \max_{k, m} \{ p_{km} \}.$$

Задача 1. Назначить операции на станки таким образом, чтобы технологический маршрут был наиболее производительным (сумма длительностей обработки и транспортировки заготовки была минимальной).

Задача 2. Назначить операции на станки таким образом, чтобы технологический маршрут был наиболее сбалансированным (длительности обработки и транспортировки заготовки были по возможности выравнены).

Варианты заданий

Вариант 1. Время выполнения операций

Операция 1	Операция 2	Операция 3	Операция 4	Операция 5
С.1 - 4	С.6 - 4	С.10 - 6	С.14 - 7	С.19 - 5
С.2 - 5	С.7 - 5	С.11 - 5	С.15 - 6	С.20 - 6
С.3 - 5	С.8 - 4	С.12 - 6	С.16 - 7	С.21 - 4
С.4 - 6	С.9 - 5	С.13 - 4	С.17 - 8	
С.5 - 6			С.18 - 7	

Вариант 2. Время выполнения операций

Операция 1	Операция 2	Операция 3	Операция 4	Операция 5
С.1 - 5	С.5 - 3	С.10 - 5	С.14 - 2	С.18 - 5
С.2 - 6	С.6 - 2	С.11 - 5	С.15 - 3	С.19 - 4
С.3 - 7	С.7 - 4	С.12 - 4	С.16 - 2	С.20 - 6
С.4 - 6	С.8 - 6	С.13 - 4	С.17 - 3	С.21 - 7
	С.9 - 4			

Вариант 3. Время выполнения операций

Операция 1	Операция 2	Операция 3	Операция 4	Операция 5
С.1 - 4	С.6 - 7	С.10 - 6	С.14 - 4	С.19 - 4
С.2 - 5	С.7 - 8	С.11 - 5	С.15 - 3	С.20 - 6
С.3 - 5	С.8 - 9	С.12 - 6	С.16 - 4	С.21 - 5
С.4 - 6	С.9 - 7	С.13 - 4	С.17 - 2	
С.5 - 6			С.18 - 6	

Вариант 4. Время выполнения операций

Операция 1	Операция 2	Операция 3	Операция 4	Операция 5
С.1 - 8	С.4 - 6	С.9 - 7	С.14 - 2	С.18 - 9
С.2 - 9	С.5 - 5	С.10 - 6	С.15 - 3	С.19 - 8
С.3 - 7	С.6 - 6	С.11 - 8	С.16 - 2	С.20 - 9
	С.7 - 7	С.12 - 9	С.17 - 3	
	С.8 - 5	С.13 - 9		

Вариант 5. Время выполнения операций

Операция 1	Операция 2	Операция 3	Операция 4	Операция 5
С.1 - 2	С.5 - 5	С.9 - 7	С.14 - 5	С.18 - 7
С.2 - 3	С.6 - 6	С.10 - 6	С.15 - 6	С.19 - 6
С.3 - 4	С.7 - 7	С.11 - 6	С.16 - 5	С.20 - 7
С.4 - 3	С.8 - 5	С.12 - 5	С.17 - 6	С.21 - 6
		С.13 - 4		

Вариант 6. Время выполнения операций

Операция 1	Операция 2	Операция 3	Операция 4	Операция 5
С.1 - 5	С.6 - 6	С.10 - 9	С.14 - 6	С.18 - 10
С.2 - 6	С.7 - 8	С.11 - 10	С.15 - 7	С.19 - 11
С.3 - 7	С.8 - 7	С.12 - 11	С.16 - 8	С.20 - 12
С.4 - 6	С.9 - 9	С.13 - 12	С.17 - 9	С.21 - 10
С.5 - 5				

Вариант 7. Время выполнения операций

Операция 1	Операция 2	Операция 3	Операция 4	Операция 5
С.1 - 8	С.5 - 5	С.10 - 7	С.14 - 7	С.18 - 10
С.2 - 9	С.6 - 6	С.11 - 8	С.15 - 8	С.19 - 12
С.3 - 9	С.7 - 7	С.12 - 7	С.16 - 9	С.20 - 11
С.4 - 10	С.8 - 6	С.13 - 10	С.17 - 10	С.21 - 12
	С.9 - 8			

Вариант 8. Время выполнения операций

Операция 1	Операция 2	Операция 3	Операция 4	Операция 5
С.1 - 4	С.6 - 4	С.10 - 6	С.14 - 7	С.19 - 5
С.2 - 7	С.7 - 5	С.11 - 5	С.15 - 6	С.20 - 6
С.3 - 5	С.8 - 6	С.12 - 7	С.16 - 5	С.21 - 4
С.4 - 6	С.9 - 5	С.13 - 4	С.17 - 8	
С.5 - 6			С.18 - 4	

Вариант 9. Время выполнения операций

Операция 1	Операция 2	Операция 3	Операция 4	Операция 5
С.1 - 2	С.4 - 6	С.9 - 5	С.14 - 6	С.18 - 5
С.2 - 3	С.5 - 5	С.10 - 6	С.15 - 5	С.19 - 6
С.3 - 3	С.6 - 6	С.11 - 4	С.16 - 7	С.20 - 4
	С.7 - 4	С.12 - 5	С.17 - 6	
	С.8 - 5	С.13 - 4		

Вариант 10. Время выполнения операций

Операция 1	Операция 2	Операция 3	Операция 4	Операция 5
С.1 - 3	С.5 - 6	С.9 - 7	С.14 - 7	С.18 - 5
С.2 - 4	С.6 - 6	С.10 - 8	С.15 - 8	С.19 - 6
С.3 - 4	С.7 - 7	С.11 - 6	С.16 - 8	С.20 - 4
С.4 - 3	С.8 - 8	С.12 - 5	С.17 - 6	С.21 - 7
		С.13 - 5		

Вариант 11. Время выполнения операций

Операция 1	Операция 2	Операция 3	Операция 4	Операция 5
С.1 - 4	С.6 - 3	С.10 - 6	С.14 - 7	С.19 - 5
С.2 - 5	С.7 - 5	С.11 - 5	С.15 - 6	С.20 - 6
С.3 - 7	С.8 - 6	С.12 - 3	С.16 - 6	С.21 - 7
С.4 - 8	С.9 - 5	С.13 - 4	С.17 - 8	
С.5 - 6			С.18 - 7	

Вариант 12. Время выполнения операций

Операция 1	Операция 2	Операция 3	Операция 4	Операция 5
С.1 – 6	С.5 – 6	С.10 – 5	С.14 – 5	С.18 – 5
С.2 – 4	С.6 – 7	С.11 – 5	С.15 – 4	С.19 – 7
С.3 – 7	С.7 – 4	С.12 – 8	С.16 – 2	С.20 – 6
С.4 – 6	С.8 – 6 С.9 – 4	С.13 – 7	С.17 – 3	С.21 – 3

Вариант 13. Время выполнения операций

Операция 1	Операция 2	Операция 3	Операция 4	Операция 5
С.1 – 4	С.6 – 5	С.10 – 6	С.14 – 3	С.19 – 7
С.2 – 5	С.7 – 8	С.11 – 5	С.15 – 6	С.20 – 6
С.3 – 6	С.8 – 9	С.12 – 3	С.16 – 5	С.21 – 5
С.4 – 6	С.9 – 7	С.13 – 4	С.17 – 6	
С.5 – 3			С.18 – 6	

Вариант 14. Время выполнения операций

Операция 1	Операция 2	Операция 3	Операция 4	Операция 5
С.1 – 8	С.4 – 6	С.9 – 5	С.14 – 6	С.18 – 5
С.2 – 5	С.5 – 5	С.10 – 6	С.15 – 3	С.19 – 8
С.3 – 7	С.6 – 4	С.11 – 8	С.16 – 5	С.20 – 4
	С.7 – 7	С.12 – 7	С.17 – 4	
	С.8 – 5	С.13 – 3		



Методические указания

Решение задачи рассмотрим на примере оптимизации техпроцесса, включающего две операции, каждая из которых может быть выполнена на одном из четырех станков (см. табл. 1.1 и 1.2).

Таблица 1.1

Время выполнения операций

	Операция 1	Операция 2
Станок 1	2	
Станок 2	3	
Станок 3	4	
Станок 4	4	
Станок 5		4
Станок 6		5
Станок 7		6
Станок 8		3

Таблица 1.2

Время транспортировки заготовок
от станка к станку

	С.5	С.6	С.7	С.8
С.1	2	3	4	5
С.2	3	4	4	3
С.3	4	4	3	3
С.4	5	4	3	2

Математическая модель задачи составляется следующим образом.

Каждому вероятному назначению i -й операции на j -й станок ставится в соответствие пара вершин орграфа, первая из которых обозначается x_{ij}^i и символизирует начало, вторая x_{ij}^j - конец i -й операции на j -м станке. Дуга (x_{ij}^i, x_{ij}^j) нагружается временем выполнения рассматриваемой операции на этом станке (см. рис.1.1).

Далее каждая из вершин x_{ij}^k соединяется дугами $(x_{ij}^k, x_{i+1,r}^n)$ с каждой из вершин $x_{i+1,r}^n$. Каждая из дуг $(x_{ij}^k, x_{i+1,r}^n)$ нагружается временами транспортировки заготовки от s -го станка к r -му. Затем вводится фиктивная вершина s , символизирующая начало техпроцесса, и соединяется дугами $(s, x_{i,r}^n)$, нагруженными нулями, с каждой из начальных вершин $x_{i,r}^n$, первой операции. Наконец вводится фиктивная вершина t , символизирующая конец техпроцесса, и соединяется дугами $(x_{n,i}^k, t)$, нагруженными нулями, с каждой из конечных

вершин x_{ns}^k последней операции (рис 1.2). В результате построен граф, каждый путь которого из вершины s в вершину t соответствует одному из вариантов назначения операций на станки.

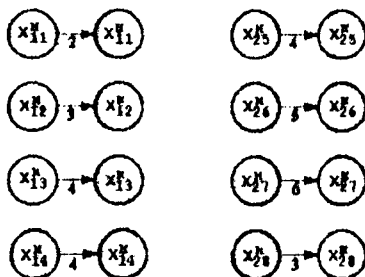


Рис. 1.1. Отображение вариантов назначения операций на станки в математической модели

Очевидно, кратчайший из всевозможных путей соответствует наиболее производительному, тончайший - наиболее сбалансированному технологическому процессу.

На рис.1.3 продемонстрирована работа алгоритма поиска кратчайшего, на рис 1.4 - тончайшего пути в графе, представленном на рис 1.2.

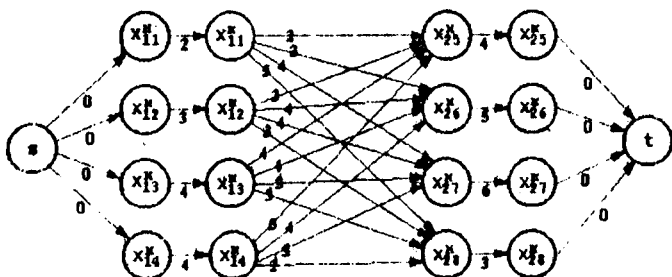


Рис 1.2. Математическая модель задачи оптимизации технологического маршрута

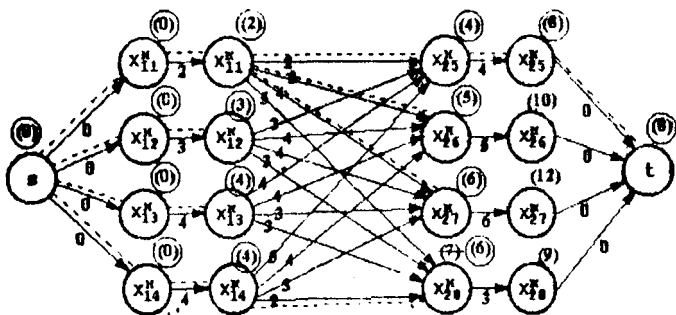


Рис. 1.3. Поиск наиболее производительного техпроцесса

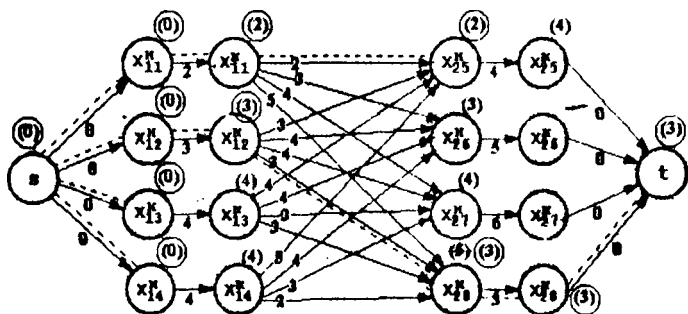
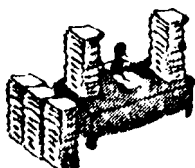


Рис. 1.4. Поиск наиболее сбалансированного техпроцесса



Содержание отчета

1. Название работы.
2. Постановка задачи.
3. Математическая модель технологического процесса.

4. Кратчайший путь в графе. Наиболее производительный технологический маршрут.
5. Тончайший путь в графе. Наиболее сбалансированный технологический маршрут.

Контрольные вопросы

1. Дать определение графа, ребра, дуги, пути.
2. Дать определение взвешенного графа, дерева, корневого дерева.
3. Дать определение веса, длины, тонкости, ширины графа и пути.
4. Изложить алгоритм поиска кратчайшего пути в графе.
5. Изложить алгоритм поиска тончайшего пути в графе.
6. Изложить методику построения математической модели задачи оптимизации технологического маршрута.

Лабораторная работа № 2

ОПТИМИЗАЦИЯ ОСНАЩЕНИЯ МАГАЗИНА ОБРАБАТЫВАЮЩЕГО ЦЕНТРА



Цель работы

Овладение навыками разработки математической модели и решения задачи оптимизации оснащения магазина обрабатывающего центра.

Постановка задачи

Имеется n различных видов инструмента для оснащения магазина обрабатывающего центра, причем число инструментов каждого вида можно считать неограниченным. Известно, что каждый инструмент i -го вида занимает a_i гнезд обрабатывающего центра и время его работы до переточки равно c_i . После установки по одному инструменту каждого вида осталось b свободных гнезд обрабатывающего центра. Необходимо оснастить оставшуюся свободной часть магазина таким образом, чтобы суммарное время работы инструментов было максимальным.

Варианты заданий

Вариант 1

Инструменты	1	2	3	4	5	6
Стойкость	1	3	1	2	3	3
Число занимаемых гнезд	1	2	3	2	5	4

Вариант 2

Инструменты	1	2	3	4	5	6
Стойкость	c_i 1	4	4	2	1	3
Число занимаемых гнезд	a_i 1	2	3	4	3	4

Вариант 3

Инструменты	1	2	3	4	5	6
Стойкость	2	3	1	4	2	1
Число занимаемых гнезд	5	4	3	3	2	1

Вариант 4

Инструменты	1	2	3	4	5	6
Стойкость	2	3	5	1	3	3
Число занимаемых гнезд	4	2	3	1	5	4

Вариант 5

Инструменты	1	2	3	4	5	6
Стойкость	2	3	4	6	1	1
Число занимаемых гнезд	4	2	3	4	5	1

Вариант 6

Инструменты	1	2	3	4	5	6
Стойкость	2	4	1	1	2	7
Число занимаемых гнезд	3	2	3	1	5	4

Вариант 7

Инструменты	1	2	3	4	5	6
Стойкость	2	3	4	5	3	1
Число занимаемых гнезд	2	2	3	4	5	1

Вариант 8

Инструменты	1	2	3	4	5	6
Стойкость	2	3	7	1	9	8
Число занимаемых гнезд	5	2	3	1	5	4

Вариант 9

Инструменты	1	2	3	4	5	6
Стойкость	2	7	6	2	8	5
Число занимаемых гнезд	4	4	3	1	5	2

Вариант 10

Инструменты	1	2	3	4	5	6
Стойкость	2	4	6	5	7	1
Число занимаемых гнезд	1	2	3	4	5	1

Вариант 11

Инструменты	1	2	3	4	5	6
Стойкость	1	7	4	1	5	3
Число занимаемых гнезд	1	4		1	3	4

Вариант 12

Инструменты	1	2	3	4	5	6
Стойкость	3	7	5	1	8	9
Число занимаемых гнезд	1	4	2	1	3	5

Вариант 13

Инструменты	1	2	3	4	5	6
Стойкость	2	7	3	1	4	3
Число занимаемых гнезд	1	4	2	1	3	4

Инструменты	1	2	3	4	5	6
Стойкость	3	6	7	2	8	9
Число занимаемых гнезд	1	4	2	1	3	6

Для всех вариантов число свободных гнезд магазина обрабатывающего центра равно 7.



Методические указания

Математическая модель задачи составляется следующим образом.

Обозначим через $x_i, i = \overline{1, n}$, искомое число инструментов i -го вида. Тогда поставленную задачу можно формализовать следующим образом:

$$z = c_1 x_1 + c_2 x_2 + \dots + c_n x_n \rightarrow \max ;$$

$$a_1 x_1 + a_2 x_2 + \dots + a_n x_n = b.$$

Для решения этой задачи вначале следует проверить, имеются ли различные инструменты, занимающие одно и то же число гнезд. Если существуют номера инструментов u и v такие, что $a_u = a_v$, а $c_u \leq c_v$, то для достижения поставленной цели необходимо инструменты u -го вида из задачи исключить, т.е. принять $x_u := 0$. Выполнив необходимое число таких операций, получим так называемую приведенную задачу, где все $a_i, i = \overline{1, n}$, различны.

Далее строится оргграф, содержащий $(b + 1)$ вершину с номерами $0, 1, \dots, b$. Две вершины i и k в этом графе соединяются дугой (i, k) в том случае, если можно найти номер инструмента j такой, что $a_j = k - i$. Такую дугу нагружаем стойкостью c_j этого инструмента. Таким образом, указанная дуга идентифицирует собой j -й инструмент.

Любой путь из вершины 0 в вершину b в этом графе представляет один из вариантов оснащения магазина обрабатывающего центра инструментами, а длина этого пути равна суммарной стойкости соответствующих ему инструментов. Следовательно, для решения поставленной задачи инструментов j -го вида необходимо брать столько, сколько содержится в длиннейшем пути из вершины 0 в вершину b дуг (i, k) со свойством $k - i = a_j$.

Пример:

Инструменты	1	2	3	4	5
Стойкость	1	3	4	5	1
Число занимаемых гнезд	2	4	2	3	1

Решение:

$$z := x_1 + 3x_2 + 4x_3 + 5x_4 + x_5 \rightarrow \max;$$

$$x_1 + 4x_2 + 2x_3 + 3x_4 + x_5 = 5.$$

Находим, что $\alpha_1 = \alpha_3$. Так как $c_1 < c_3$, то $x_1 := 0$ и задачу приводим к виду

$$z := 3x_2 + 4x_3 + 5x_4 + x_5 \rightarrow \max;$$

$$4x_2 + 2x_3 + 3x_4 + x_5 = 5.$$

Строим оргграф и находим длиннейший путь из вершины 0 в вершину 5. Длиннейшим является следующий путь: $(0,3), (3,5)$ (см. рис. 2.1).

Для 1-й дуги этого пути разность номеров вершин равна 3. Так как $a_4 = 3$, то эта дуга соответствует инструменту номер 4. Вторая дуга маршрута соответствует инструменту, занимающему 2 гнезда, следовательно, $x_3 = 1$.

Ответ: $x_1 = x_2 = x_5 = 0$, $x_3 = 1$, $x_4 = 1$, другими словами, магазин следует оснастить одним инструментом с номером три и одним с номером 4.

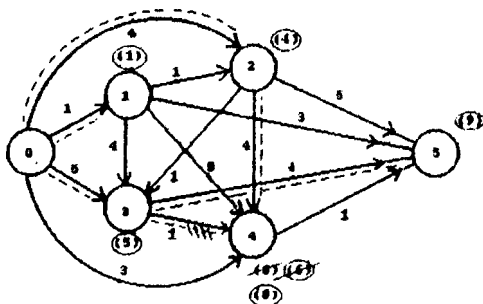
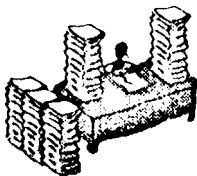


Рис. 2.1. Решение задачи об оснащении магазина обрабатывающего центра



Содержание отчета

1. Название работы.
2. Постановка задачи.
3. Математическая модель задачи.
4. Длиннейший путь в графе.
5. Оптимальное оснащение магазина.

Контрольные вопросы

1. Изложить алгоритм поиска длиннейшего пути в графе.
2. Изложить методику построения математической модели задачи оптимизации оснащения магазина обрабатывающего центра.
3. Каким образом задача может быть решена без привлечения алгоритма поиска длиннейшего пути в графе?

Лабораторная работа № 3

ОПТИМИЗАЦИЯ ПОСЛЕДОВАТЕЛЬНОСТИ ПЕРЕНАЛАДОК ТЕХНОЛОГИЧЕСКОЙ ЛИНИИ



Цель работы

Овладение навыками использования метода ветвей и границ для решения технологических задач.

Постановка задачи

Для обработки на технологической линии поступило 7 партий заготовок. При переходе от обработки одной партии к обработке следующей необходимо выполнять переналадку технологической линии, для обработки всех партий необходимо 6 таких переналадок. Задача состоит в определении такого порядка запуска партий заготовок на обработку, при котором суммарное время переналадок было бы минимальным.

Варианты заданий

Вариант 1

	1	2	3	4	5	6	7
1	∞	13	14	6	8	4	12
2	12	∞	12	16	15	7	7
3	11	10	∞	8	8	5	17
4	8	8	5	∞	10	9	8
5	6	8	9	7	∞	4	9
6	7	6	15	14	16	∞	10
7	9	14	8	15	7	12	∞

Вариант 2

	1	2	3	4	5	6	7
1	∞	8	9	7	8	9	6
2	12	∞	11	4	7	4	6
3	3	5	∞	6	7	4	7
4	6	2	3	∞	4	9	9
5	7	4	5	7	∞	8	8
6	6	3	9	5	7	∞	10
7	5	5	8	12	11	11	∞

Вариант 3

	1	2	3	4	5	6	7
1	∞	11	10	17	6	8	9
2	9	∞	14	15	7	8	9
3	7	9	∞	5	8	4	5
4	6	10	7	∞	5	8	6
5	15	12	6	9	∞	7	7
6	7	14	7	8	7	∞	6
7	9	11	7	8	10	5	∞

Вариант 4

	1	2	3	4	5	6	7
1	∞	7	8	9	7	9	6
2	13	∞	8	14	8	8	7
3	14	5	∞	6	7	5	6
4	4	13	4	∞	13	14	5
5	6	11	5	12	∞	17	4
6	17	7	7	11	2	∞	10
7	5	6	6	16	8	9	∞

Вариант 5

	1	2	3	4	5	6	7
1	∞	6	7	8	13	14	8
2	7	∞	5	6	8	9	15
3	13	14	∞	8	9	9	6
4	12	7	4	∞	5	5	7
5	11	6	3	3	∞	14	8
6	14	5	12	14	11	∞	9
7	3	14	11	6	7	8	∞

Вариант 6

	1	2	3	4	5	6	7
1	∞	11	9	11	12	6	7
2	6	∞	12	8	16	13	5
3	15	7	∞	13	7	15	4
4	13	14	8	∞	14	6	4
5	6	12	14	9	∞	5	5
6	8	7	13	12	8	∞	6
7	7	9	8	2	11	7	∞

Вариант 7

	1	2	3	4	5	6	7
1	∞	9	5	6	12	3	9
2	13	∞	14	7	14	4	8
3	14	8	∞	8	15	5	9
4	11	7	13	∞	16	6	12
5	10	6	11	9	∞	7	11
6	12	5	2	9	6	∞	8
7	9	14	13	8	7	9	∞

Вариант 8

	1	2	3	4	5	6	7
1	∞	11	10	12	6	7	8
2	16	∞	5	6	11	8	8
3	11	10	∞	11	5	6	7
4	6	8	13	∞	7	6	8
5	5	7	8	9	∞	5	7
6	14	6	7	14	11	∞	6
7	6	15	6	8	10	7	∞

Вариант 9

	1	2	3	4	5	6	7
1	∞	6	13	5	6	6	7
2	11	∞	12	4	7	8	15
3	12	9	∞	11	10	9	13
4	11	12	3	∞	13	13	7
5	13	13	14	6	∞	15	9
6	14	12	11	10	13	∞	8
7	5	6	7	8	9	7	∞

Вариант 10

	1	2	3	4	5	6	7
1	∞	5	6	11	10	8	9
2	4	∞	7	8	13	10	7
3	5	9	∞	9	8	12	7
4	6	6	7	∞	7	8	8
5	7	9	9	4	∞	6	7
6	8	8	10	11	6	∞	5
7	5	9	7	4	5	6	∞

Вариант 11

	1	2	3	4	5	6	7
1	∞	14	15	6	7	8	14
2	13	∞	5	8	9	8	11
3	5	14	∞	7	8	7	15
4	14	6	6	∞	7	6	11
5	6	7	12	3	∞	8	10
6	5	8	13	4	6	∞	9
7	7	5	12	13	5	4	∞

Вариант 12

	1	2	3	4	5	6	7
1	∞	9	8	7	7	8	9
2	5	∞	6	7	6	5	4
3	8	9	∞	5	12	4	5
4	5	8	7	∞	6	5	4
5	6	8	4	2	∞	11	12
6	13	6	6	6	5	∞	10
7	4	5	8	5	4	9	∞

Вариант 13

	1	2	3	4	5	6	7
1	∞	12	7	9	13	8	6
2	14	∞	5	8	9	8	11
3	5	9	∞	7	8	7	5
4	4	6	6	∞	7	6	9
5	6	7	12	3	∞	8	10
6	5	8	10	4	6	∞	7
7	7	5	11	13	5	4	∞

Вариант 14.

	1	2	3	4	5	6	7
1	∞	9	8	7	7	8	9
2	5	∞	6	7	6	5	4
3	8	9	∞	5	13	15	5
4	5	8	7	∞	6	5	4
5	6	8	15	2	∞	11	12
6	3	6	6	6	5	∞	8
7	14	5	8	5	12	9	∞



Методические указания

Рассмотрим задачу для случая обработки четырех партий заготовок, время переналадки технологической линии для которых указано в табл. А.

Таблица А

$i \setminus j$	1	2	3	4	α_i
1	∞	9	8	4	4
2	7	∞	4	5	4
3	5	3	∞	6	3
4	4	7	1	∞	1

Решение. Приводим таблицу по строкам. Константы приведения записываем справа в дополнительном столбце. Приведенная по строкам таблица имеет вид "А.". Эту таблицу приводим по столбцам. Константы приведения записываем снизу в дополнительной строке. Полностью приведенная таблица имеет вид A^0 , а нижняя граница $\phi^0 = \gamma = 12 + 2 = 14$.

Таблица A' .

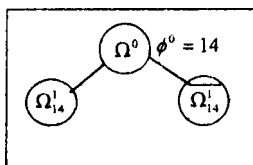
$i \setminus j$	1	2	3	4
1	∞	5	4	0
2	3	∞	0	1
3	2	0	∞	3
4	3	6	0	∞
β_j	2	0	0	0

Таблица A^0 .

$i \setminus j$	1	2	3	4
1	∞	5	4	0(5)
2	1	∞	0(1)	1
3	0(1)	0(3)	∞	3
4	1	6	0(1)	∞

Найдем дугу, относительно которой наиболее выгодно разбить множество Ω^0 .

С этой целью каждый нуль в таблице A^0 мысленно заменяем на ∞ , находим сумму $\overline{\gamma}_{ij}' = \alpha_i' + \beta_j'$ минимального элемента α_i' в строке и минимального элемента β_j' в столбце, на пересечении которых этот нуль расположен, и записываем эту сумму рядом с нулем в скобках. Наибольшая из сумм $\overline{\gamma}_{14}' = 5$ соответствует дуге (x_1, x_4) . Следовательно, множество Ω^0 разбиваем на Ω_{14}^1 и $\overline{\Omega}_{14}^1$.



Составляем табл. A_{14}^1 и $A_{\overline{14}}^1$, соответствующие этим множествам.

Таблица A_{14}^1 .

$i \setminus j$	1	2	3	α_i
2	1	∞	0(1)	0
3	0(1)	0(6)	∞	0
4	∞	6	0(6)	0
β_j	0	0	0	

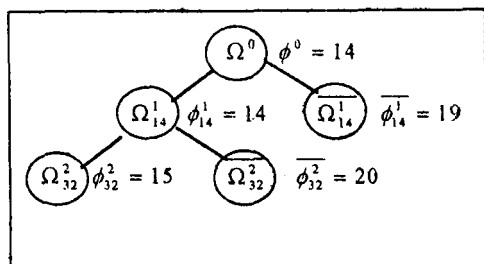
Таблица $A_{\overline{14}}^1$.

$i \setminus j$	1	2	3	4
1	∞	5	4	∞
2	1	∞	0	1
3	0	0	∞	3
4	1	6	0	∞

Для получения нижней границы $\overline{\phi}_{14}^1$ приводим табл. A_{14}^1 (достаточно привести ее по первой строке и четвертому столбцу). $\overline{\phi}_{14}^1 = \phi^0 + \overline{\gamma}_{14}' = 14 + 5 = 19$. Для

определения нижней границы ϕ_{14}^1 дополнительно приводим табл. A_{14}^1 . Получаем $\gamma_{14}^1 = 0 + 0 = 0$. $\phi_{14}^1 = \phi^0 + \gamma_{14}^1 = 14 + 0 = 14$.

Из сравнения $\overline{\phi}_{14}^1$ и ϕ_{14}^1 следует, что дальнейшему ветвлению подлежит множество Ω_{14}^1 .



Находим дугу для разбиения этого множества. С этой целью анализируем табл. A_{14}^1 : элементы $a_{ij} = 0$ мысленно заменяем на ∞ , находим суммы минимальных элементов в соответствующей строке и столбце и выбираем среди них максимальную. В результате в качестве разбивающей выбираем дугу (x_3, x_2) и множество Ω_{14}^1 разбиваем на подмножества Ω_{32}^2 и $\overline{\Omega}_{32}^2$.

Для множества $\overline{\Omega}_{32}^2$ после дополнительного приведения получаем следующую табл. \overline{A}_{32}^2 . При этом $\overline{\gamma}_{32}^2 = 6$, $\overline{\phi}_{32}^2 = \phi_{14}^1 + \overline{\gamma}_{32}^2 = 20$. Для Ω_{32}^2 после дополнительного приведения получаем A_{32}^2 , при этом $\gamma_{32}^2 = 1$, $\phi_{32}^2 = 14 + 1 + 15$.

Таблица A_{32}^2

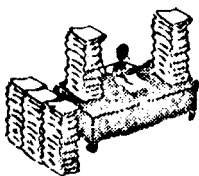
$i \setminus j$	1	3
2	0	∞
4	∞	0

Таблица \overline{A}_{32}^2

$i \setminus j$	1	2	3
2	1	∞	0
3	0	∞	∞
4	∞	0	0

Так как A_{32}^2 является таблицей размером 2×2 , то Ω_{32}^2 включает один единственный контур. Из дерева видно, что он включает дуги (x_1, x_4) и (x_3, x_2) .

Из табл. \overline{A}_{32}^2 видно, что в него входят также дуги (x_2, x_1) и (x_4, x_3) . Получаем следующую последовательность обработки партий заготовок: $1 \Rightarrow 4 \Rightarrow 3 \Rightarrow 2 \Rightarrow 1$. Длина этого маршрута равна $\rho \in c = 15$. Длина этого маршрута меньше нижних границ для всех оборванных ветвей дерева, следовательно, этот маршрут является решением задачи.



Содержание отчета

1. Название работы.
2. Постановка задачи.
3. Решение задачи.
4. Последовательность обработки партий заготовок.

Контрольные вопросы

1. Сущность метода ветвей и границ.
2. Алгоритм Литтла.
3. Дать определение контура, гамильтонова контура.
4. Дать определение нижней границы, точной нижней границы (инфинума) числового множества.
5. Почему сумма констант приведения матрицы является нижней границей множества длин гамильтоновых контуров?
6. Почему в качестве разбивающей принимается дуга нулевой длины?
7. Почему среди нулевых в качестве разбивающей принимается дуга, сумма дополнительных констант приведения для которой максимальна?

Лабораторная работа № 4

ОПТИМИЗАЦИЯ ОБРАБОТКИ НА СТАНКЕ С ЧПУ



Цель работы

Овладение навыками использования метода ветвей и границ для решения технологических задач.

Постановка задачи

На станке с ЧПУ требуется обработать 6 отверстий в корпусной детали. Положения осей отверстий и исходное положение инструмента заданы на чертеже. Найти такую последовательность обработки отверстий, чтобы суммарная длина холостых перемещений инструмента была минимальной.

Варианты заданий

Вариант 1	Рис. 4.1, положение инструмента 1
Вариант 2	Рис. 4.1, положение инструмента 2
Вариант 3	Рис. 4.2, положение инструмента 1
Вариант 4	Рис. 4.2, положение инструмента 2
Вариант 5	Рис. 4.2, положение инструмента 3
Вариант 6	Рис. 4.3, положение инструмента 1
Вариант 7	Рис. 4.3, положение инструмента 2
Вариант 8	Рис. 4.3, положение инструмента 3
Вариант 9	Рис. 4.4, положение инструмента 1
Вариант 10	Рис. 4.4, положение инструмента 2
Вариант 11	Рис. 4.5, положение инструмента 1
Вариант 12	Рис. 4.5, положение инструмента 2
Вариант 13	Рис. 4.5, положение инструмента 3
Вариант 14	Рис. 4.5, положение инструмента 4

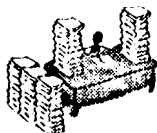
Методические указания



Эта задача с достаточной очевидностью сводится к задаче о коммивояжере. В роли городов здесь будут выступать отверстия и исходное положение инструмента, в роли коммивояжера – инструмент. Расстояния между отверстиями и исходным положением инструмента вычисляются на основании данных чертежа. Найденный в результате решения задачи оптимальный циклический маршрут коммивояжера разрывается таким образом, чтобы отправной точкой для инструмента являлось его исходное положение.

Работа алгоритма Литтла решения задачи о коммивояжере подробно рассмотрена в лабораторной работе №3.

Содержание отчета



1. Название работы.
2. Постановка задачи.
3. Эскиз детали.
4. Последовательность обработки отверстий.

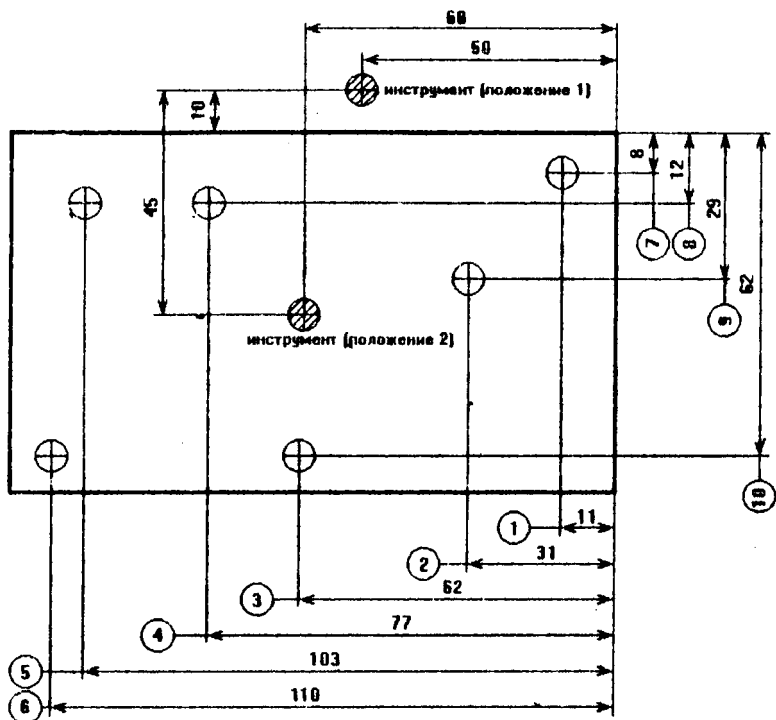


Рис. 4.1. Операция 005. Сверлить отверстия, выдерживая размеры 1...10

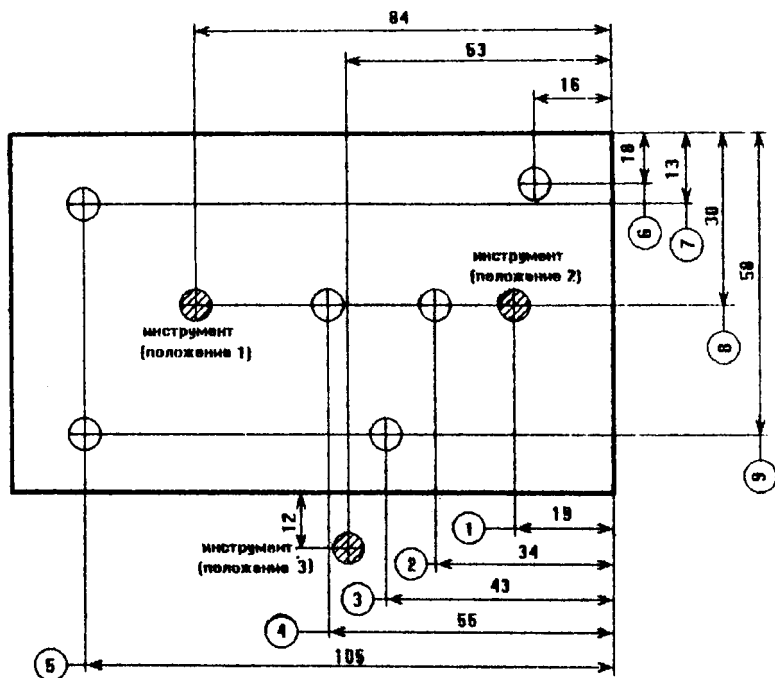


Рис. 4.2. Операция 010. Сверлить отверстия, выдерживая размеры 1...9

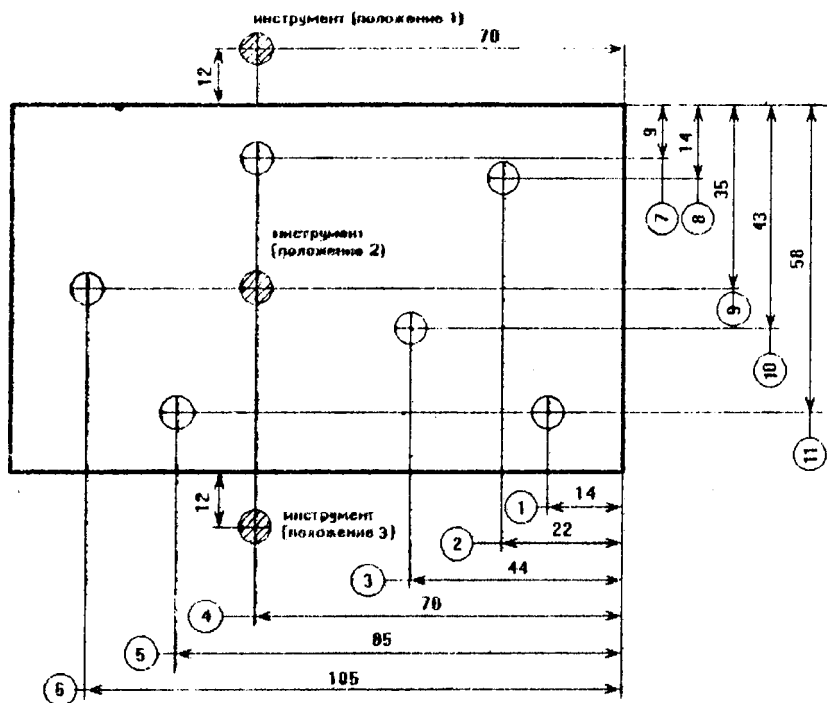


Рис. 4.3 Операция 015. Сверлить отверстия, выдерживая размеры 1...11

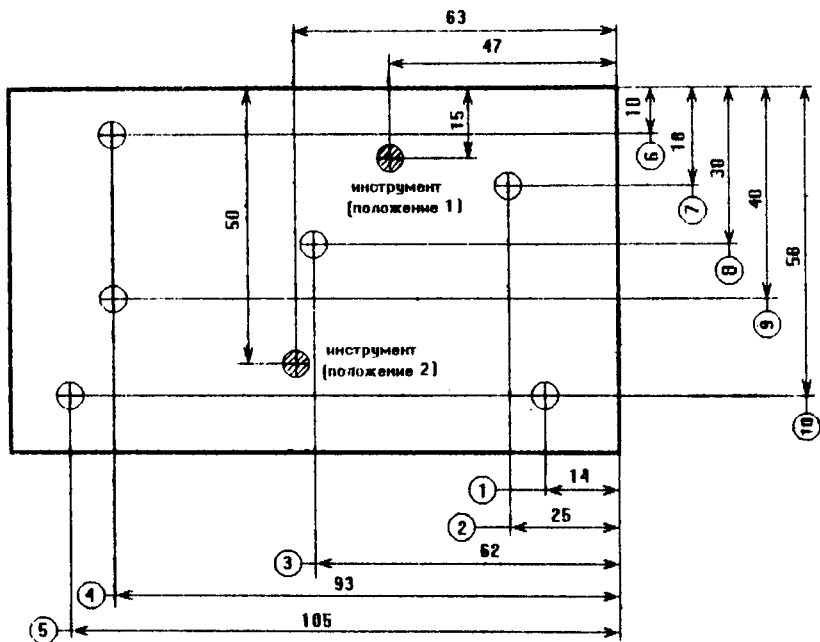


Рис. 4.4. Операция 020. Сверлить отверстия, выдерживая размеры 1...10

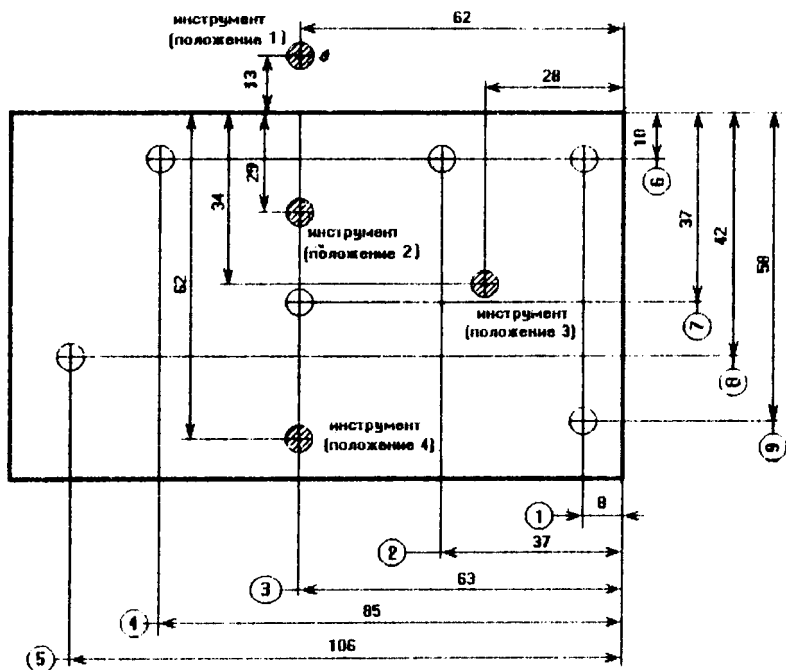


Рис. 4.5 Операция 025 Сверлить отверстие, выдерживая размеры 1..9

Контрольные вопросы

1. Сущность метода ветвей и границ.
2. Алгоритм Литтла.
3. Дать определение контура, гамильтонова контура.
4. Дать определение нижней границы числового множества.
5. Почему сумма констант приведения матрицы является нижней границей множества длин гамильтоновых контуров?
6. Почему в качестве разбивающей принимается дуга нулевой длины?
7. Почему среди нулевых в качестве разбивающей принимается дуга, сумма дополнительных констант приведения для которой максимальна?

Лабораторная работа № 5

ОПТИМИЗАЦИЯ ПОСЛЕДОВАТЕЛЬНОСТИ ГОРЯЧЕЙ ОБРАБОТКИ



Цель работы

Овладение навыками использования метода ветвей и границ для решения технологических задач

Основные положения

Горячая обработка давлением отличается тем свойством, что каждая деталь может ожидать обработку только перед первым станком линии, а перед остальными ожидание недопустимо. В данной работе рассматриваются только компактные расписания горячей обработки, которые отличаются тем, что при выбранной последовательности сократить время обработки невозможно. На рис. 5.1 приведен пример такого расписания в виде графика Ганта загрузки станков.

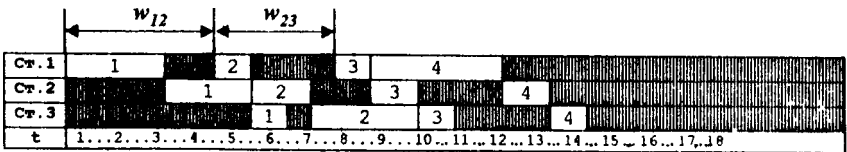


Рис. 5.1. Расписание горячей обработки

Задача о горячей обработке сводится к известной задаче о коммивояжере, которая подробно рассмотрена в лабораторной работе «Оптимизация последовательности переналадок технологической линии». С этой целью вводится в рассмотрение фиктивная $(h+1)$ -я деталь, с нулевыми временами обработки $P_{h+1,1}, P_{h+1,2}, \dots, P_{h+1,m} = 0$, где h - количество деталей; m - станков. В качестве городов выступают детали, в качестве расстояний между ними - величины w_{ij} , равные временам между моментами начала обработки i -й и j -й деталей при условии, что j -я деталь обрабатывается сразу за i -й (см. рисунок). Эти величины вычисляются по формуле

$$w_{ij} = \max_{1 \leq k \leq m} \left\{ \sum_{s=1}^k p_{is} - \sum_{s=1}^{k-1} p_{js} \right\}.$$

Очевидно, $w_{i,h+1} = p_{i1} + p_{i2} + \dots + p_{im}$, т.е. обработка фиктивной детали начинается только после окончания всей обработки предыдущей.

Постановка задачи

На линии горячей обработки, состоящей из 4 станков, нужно обработать 5 различных деталей. Все детали должны проходить вдоль линии в одном направлении через каждый станок. Заданы длительности p_{ij} обработки детали i на j -м станке $\forall i = \overline{1,5}, \forall j = \overline{1,4}$. Требуется составить последовательность горячей обработки деталей, позволяющую закончить ее за минимальное время.

Варианты заданий

Вариант 1

$j \setminus i$	1	2	3	4	5
1	5	3	4	7	4
2	12	7	2	8	7
3	11	10	9	8	5
4	8	8	5	7	11

Вариант 2

$j \setminus i$	1	2	3	4	5
1	8	4	4	3	4
2	7	7	2	8	7
3	11	10	9	8	5
4	3	8	5	7	5

Вариант 3

$j \setminus i$	1	2	3	4	5
1	8	12	4	7	4
2	7	4	2	8	8
3	3	10	9	8	5
4	8	7	5	7	5

Вариант 4

$j \setminus i$	1	2	3	4	5
1	4	13	7	9	5
2	6	7	5	8	7
3	11	10	3	8	5
4	8	8	5	7	9

Вариант 5

j \ i	1	2	3	4	5
1	9	6	4	7	4
2	3	7	2	8	7
3	11	3	9	8	4
4	8	8	5	7	8

Вариант 7

j \ i	1	2	3	4	5
1	5	7	4	7	11
2	6	9	2	8	7
3	11	10	9	7	5
4	8	3	5	7	11

Вариант 9

j \ i	1	2	3	4	5
1	6	7	4	7	11
2	4	7	9	8	7
3	11	5	4	8	5
4	2	8	5	7	11

Вариант 11

j \ i	1	2	3	4	5
1	5	13	14	7	12
2	6	7	12	5	7
3	13	10	9	8	5
4	8	5	5	7	11

Вариант 13

j \ i	1	2	3	4	5
1	7	9	6	4	8
2	10	7	12	8	5
3	10	10	9	4	5
4	6	8	5	8	11

Вариант 6

J \ i	1	2	3	4	5
1	11	6	4	7	4
2	3	7	2	8	7
3	11	10	9	8	5
4	8	5	5	7	11

Вариант 8

j \ i	1	2	3	4	5
1	7	3	4	10	4
2	12	4	10	8	7
3	6	10	9	9	5
4	11	8	5	7	6

Вариант 10

j \ i	1	2	3	4	5
1	9	11	4	7	4
2	12	5	12	8	7
3	9	3	9	8	5
4	8	8	5	7	11

Вариант 12

j \ i	1	2	3	4	5
1	15	3	4	7	4
2	10	7	12	8	7
3	3	10	9	8	5
4	8	8	5	7	11

Вариант 14

j \ i	1	2	3	4	5
1	10	13	4	7	4
2	5	7	12	8	7
3	11	10	9	8	5
4	4	8	15	7	11

**Методические указания**

Рассмотрим пример решения задачи составления оптимального расписания горячей обработки четырех деталей на четырех станках. Времена выполнения операций приведены в следующей таблице.

	Деталь 1	Деталь 2	Деталь 3	Деталь 4
Станок 1	4	6	4	9
Станок 2	7	9	3	5
Станок 3	3	3	7	4
Станок 4	8	11	8	6

Решение. Добавляем фиктивную деталь 5 и вычисляем величины w_{ij} , $i, j = 1, \dots, 5$. Результаты вычислений заносим в таблицу, аналогичную таблице расстояний между городами в задаче о коммивояжере.

В соответствии с определением фиктивной детали

$$w_{5i} = 0, \quad \forall i = \overline{1,4};$$

$w_{i5} = \sum_{j=1}^4 p_{ij}, \quad \forall i = \overline{1,4}$, т.е. эти величины равны суммам чисел соответствующих столбцов исходной таблицы;

$$w_{15} = \max\{4, 4+7-6, 4+7+3-6-9, 4+7+3+8-6-9-3\} = 5;$$

$$w_{25} = \max\{4, 4+7-4, 4+7+3-4-3, 4+7+3+8-4-3-7\} = 8;$$

$$w_{45} = \max\{4, 4+7-9, 4+7+3-9-5, 4+7+3+8-9-5-4\} \text{ и т.д.}$$

	1	2	3	4	5
1	∞	5	8	4	22
2	15	∞	15	11	29
3	8	4	∞	4	22
4	10	8	11	∞	24
5	0	0	0	0	∞

Заполнив таблицу, решаем задачу о коммивояжере и получаем следующий его маршрут: $2 \rightarrow 4 \rightarrow 1 \rightarrow 5 \rightarrow 3 \rightarrow 2$. Возвращаясь к исходной задаче, исключением фиктивной детали разрываем приведенный цикл и получаем следующую последовательность горячей обработки деталей: $3 \rightarrow 2 \rightarrow 4 \rightarrow 1$. Составляем график Ганта загрузки станков

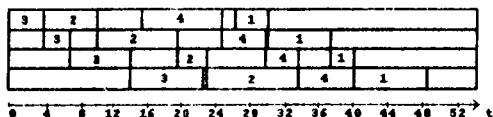
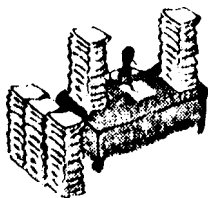


Рис. 5.2. Расписание горячей обработки



Содержание отчета

1. Название работы.
2. Постановка задачи.
3. Сведение задачи к задаче о коммивояжере.
4. Решение задачи о коммивояжере.
5. Последовательность горячей обработки.
6. График Ганта работы станков.

Контрольные вопросы

1. Понятие горячей обработки.
2. Понятие компактного расписания горячей обработки.
3. Для чего при решении задачи вводится фиктивная деталь?
4. В чем заключается физический смысл величин w_{ij} ?

Лабораторная работа № 6

МИНИМИЗАЦИЯ СУММАРНОГО ПОЛЯ ДОПУСКА

Цель работы



Овладение навыками разработки математических моделей и решения технологических задач с помощью алгоритмов поиска оптимальных остовов в графах.

Основные положения

На чертеже детали размеры, как правило, можно нанести несколькими способами. Однако более предпочтительным является тот способ, который приводит к более строгому соблюдению точностных характеристик детали при её изготовлении.

Приведем пример. На рис. 6.1 изображена деталь с пронумерованными поверхностями и показаны два варианта нанесения связывающих их размеров. Обозначим через $\Delta a, \Delta b, \Delta c, \dots, \Delta i$ допуски на размеры a, b, c, \dots, i . При первом способе нанесения размеров суммарное поле допусков равно $\Delta a + \Delta b + \Delta c + \Delta d + \Delta e$, при втором - $\Delta a + \Delta f + \Delta g + \Delta i + \Delta h$. Очевидно, суммарное поле допусков зависит от способа нанесения размеров.

Постановка задачи

Найти способ нанесения размеров для детали, эскиз которой представлен на рис. 6.4, при котором суммарное поле допусков является минимальным. Допуски на соответствующие размеры приведены в табл. 6.1.



Методические указания

С целью решения задачи каждой поверхности детали, связанной с другими размерной связью, ставится в соответствие вершина графа. Две вершины соединяются ребром в том случае, если они в принципе могут быть связаны размерной связью. Для детали, приведенной на рис. 6.1, этот граф примет вид, показанный на рис. 6.2.

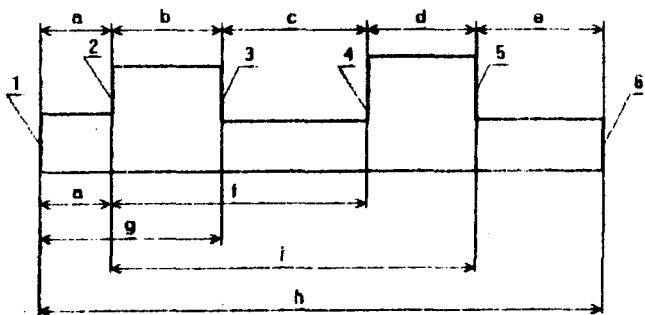


Рис. 6.1. Эскиз детали

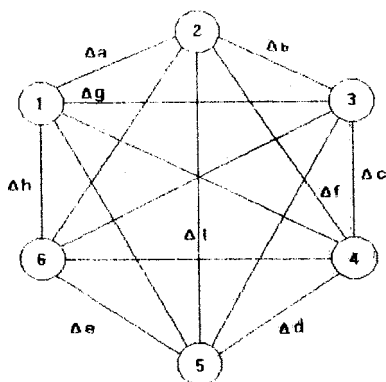


Рис. 6.2. Граф размерных связей

Ребра нагружаются допусками на соответствующие размеры (на рисунке показаны допуски только на те размеры, для которых введены обозначения). При выборе способа нанесения размеров обязательно выполняются следующие условия: во-первых, размерной связью связывается каждая из выделённых поверхностей, во-вторых, размерная цепь не содержит замкнутых участков. Следовательно, каждому из вариантов нанесения размеров соответствует один из остовов построенного графа. Двум способам нанесения размеров, показанным на рис. 6.1, соответствуют остовы построенного графа, представленные на рис. 6.3. Так как вес каждого из таких остовов равен суммарному полю допуска, то задача свелась к поиску минимального остова.

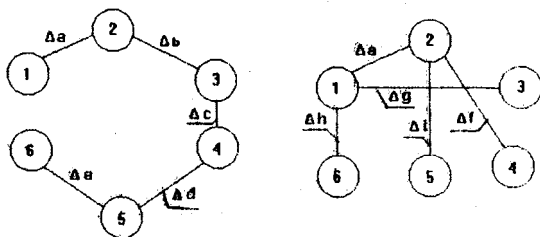


Рис. 6.3. Остовы графа размерных связей

Для поиска минимального остова необходимо воспользоваться *жадным* алгоритмом или алгоритмом ближайшего соседа.

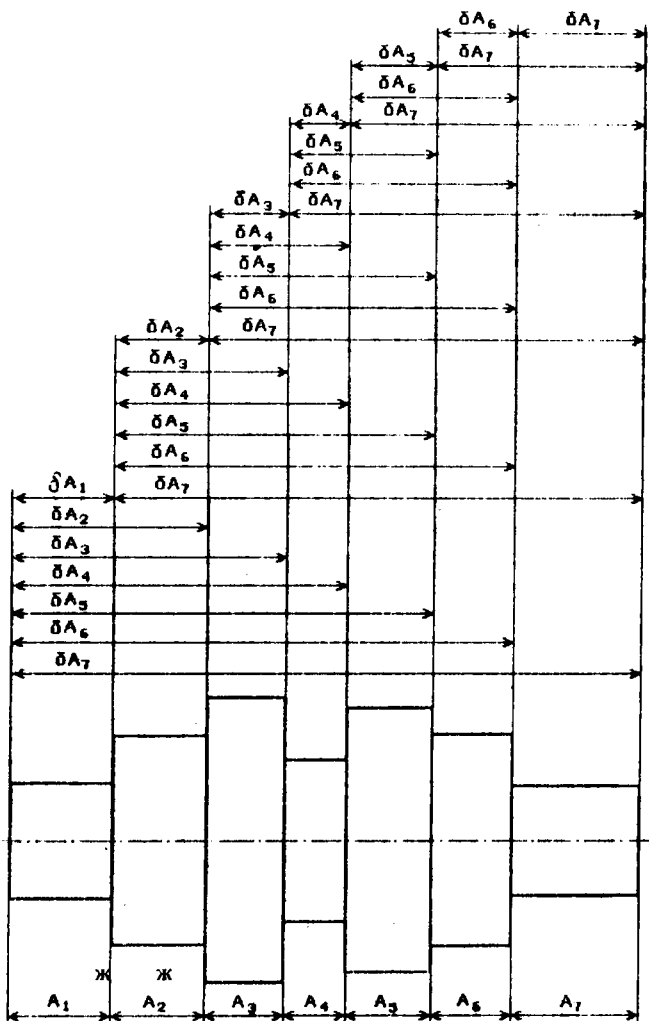


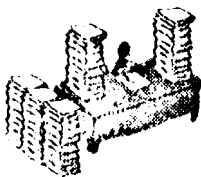
Рис. 6 4. Эскиз детали: A_i ; $i = \overline{1,7}$, - номинальные размеры звеньев размерной цепи; δA_i - допуски на размеры

Варианты записей

Таблица 6.1

Численные значения полей допуска

		Номер варианта													
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	
ΔA	0,01	0,005	0,1	0,09	0,14	0,17	0,009	0,1	0,2	0,1	0,12	0,005	0,13	0,17	
Δa	0,15	0,007	0,11	0,1	0,15	0,11	0,006	0,15	0,21	0,13	0,15	0,008	0,1	0,13	
Δa	0,12	0,01	0,14	0,11	0,11	0,13	0,004	0,17	0,19	0,12	0,17	0,007	0,12	0,11	
Δa	0,14	0,008	0,13	0,08	0,1	0,14	0,008	0,16	0,22	0,09	0,13	0,004	0,11	0,14	
Δa	0,013	0,004	0,12	0,12	0,16	0,12	0,01	0,12	0,18	0,11	0,14	0,01	0,14	0,12	
Δa	0,011	0,006	0,15	0,1	0,12	0,15	0,007	0,14	0,24	0,08	0,11	0,009	0,13	0,15	
Δa	0,017	0,009	0,17	0,13	0,13	0,11	0,005	0,13	0,2	0,12	0,1	0,006	0,15	0,01	



Содержание отчета

1. Название работы.
2. Постановка задачи.
3. Эскиз детали.
4. Математическая модель задачи
5. Оптимальный остов.
6. Оптимальный способ нанесения размеров.

Контрольные вопросы

1. Что такое остов графа, минимальный остов?
2. Алгоритм поиска минимального остова.

Лабораторная работа № 7

ОПТИМИЗАЦИЯ РАСКРОЯ ПРОМЫШЛЕННЫХ МАТЕРИАЛОВ



Цель работы

Овладение навыками использования метода линейного программирования для решения технологических задач.

Основные положения

Некоторые предприятия потребляют большое количество листового металлопроката. При разделении листов на заготовки образуются отходы, размеры которых зависят от того, как расположены заготовки на листе. В таких случаях актуальной является задача оптимального раскроя промышленного материала, которая формулируется следующим образом: каких размеров и в каких количествах следует поставлять заводу листовой прокат и как его раскраивать, чтобы обеспечить снижение отходов металла до минимума. Для решения этой задачи в общем случае должны быть установлены следующие исходные данные:

- количество n типов листов, которые могут быть поставлены предприятию, и их размеры;

- вес ρ_j одного листа j -го типа, $j = \overline{1, n}$;
- количество m типоразмеров заготовок, необходимых для производства деталей, и их размеры;
- количество L_i заготовок i -го наименования, которое необходимо изготовить за планируемый период, $i = \overline{1, m}$;
- минимальная транзитная норма C_m ;
- суммарный T фонд рабочего времени заготовительного участка предприятия.

Кроме этого для каждого типа j листа необходимо установить допустимые схемы его раскроя k_j , и для каждой такой схемы найти:

- l_{ik_j} - количество заготовок i -го наименования, которое можно получить при раскрое одного листа j -го типа k_j -м способом;
- t_{ik_j} - время, затрачиваемое на раскрой одного листа j -го типа k_j -м способом;
- c_{ik_j} - величина отхода материала, получаемого при раскрое одного листа j -го типа k_j -м способом.

С учетом введенных обозначений можно сказать, что для решения поставленной производственной задачи требуется найти количество x_{k_j} листов j -го типа, раскраиваемых k_j -м способом для всех $j = \overline{1, n}$ и для всех возможных схем раскроя.

Искомые величины x_{k_j} должны удовлетворять следующим условиям:

а) полное обеспечение потребности в заготовках

$$\sum_i \sum_{k_j} l_{ik_j} x_{k_j} = L_i, \quad i = \overline{1, m};$$

б) учет ограниченности производственных возможностей заготовительного участка

$$\sum_j \sum_{k_j} t_{k_j} x_{k_j} \leq T;$$

в) требование соблюдения транзитных норм

$$\sum_j \rho_j x_{k_j} \geq c_m, \quad j = \overline{1, n};$$

г) условие неотрицательности

$$x_{k_j} \geq 0, \forall j, \forall k_j.$$

При этом необходимо обеспечить минимум линейной функции

$$z = \sum_j \sum_{k_j} c_{k_j} x_{k_j},$$

выражающей суммарный отход материала. Получена задача линейного программирования, для решения которой необходимо воспользоваться симплекс-методом.

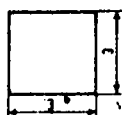
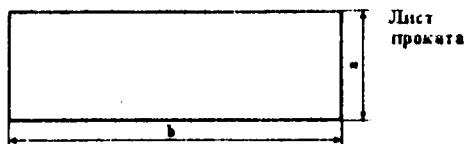
Постановка задачи

Из листового проката двух (одного) типов необходимо вырезать некоторое количество заготовок для производства 90 штук изделий. Для одного изделия требуется деталей первого типа – 3 штуки, второго – 2, третьего (если это предусмотрено в варианте задания) – 1. Возможности заготовительного участка не ограничены. Размеры листов проката и типы заготовок указаны в вариантах заданий и на рис. 7.1. При решении задачи разработать не менее двух вариантов раскроя для каждого из двух типов листового проката или не менее четырех, если прокат только одного типа.

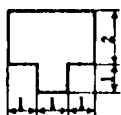
Варианты заданий

Вариант	Лист проката 1		Лист проката 2		Типы заготовок		
	а	б	а	б	Тип 1	Тип 2	Тип 3
1	2	3	4	5	6	7	8
1	4	13	5	11	А	Г	-
2	4	13	5	15	А	В	-
3	4	14	4	8	А	Е	-
4	4	11	4	7	А	Ж	-
5	4	15	4	9	А	З	-
6	4	13	5	10	Б	И	-
7	4	12	5	10	В	И	-
8	3	11	5	9	Г	И	-
9	4	14	5	9	Д	И	-
10	4	14	5	10	Е	И	-

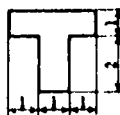
1	2	3	4	5	6	7	8
11	3	11	4	9	Г	Ж	-
12	3	12	4	8	Е	Ж	-
13	5	12	-	-	А	Б	В
14	4	13	-	-	А	Б	Г
15	3	14	-	-	А	Б	З
16	3	15	-	-	А	Б	Г
17	4	15	-	-	А	Б	Ж
18	5	15	-	-	А	Б	Е
19	5	14	-	-	А	В	Ж
20	4	15	-	-	И	В	Е
21	5	13	-	-	И	В	Г
22	4	14	-	-	И	В	Е



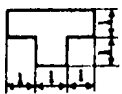
Дет. А



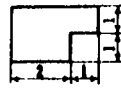
Дет. Б



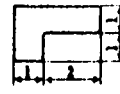
Дет. В



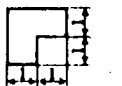
Дет. Г



Дет. Д



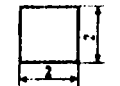
Дет. Е



Дет. Ж



Дет. З



Дет. И

Рис. 7.1. Эскизы листа проката и заготовок



Методические указания

Методику решения задачи рассмотрим на следующем примере.

Необходимо из листового проката вырезать некоторое количество заготовок двух типов А и Б для производства 90 штук изделий. Для одного изделия требуется 2 детали типа А и 10 деталей типа Б. Возможности заготовительного участка не ограничены. Размеры листа и заготовок представлены на рис. 7.2.

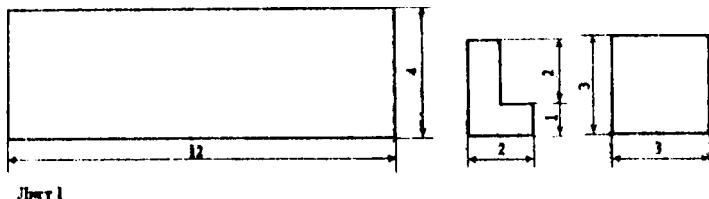


Рис. 7.2. Эскизы листа проката и заготовок

Выбираем возможные схемы раскроя листа. При этом следует иметь в виду, что в совокупности они должны обеспечивать получение заготовок всех требуемых видов, хотя конкретный вариант раскроя какого-либо листа может содержать заготовки, например, только одного вида. Не следует также допускать схем раскроя, когда размеры отхода превосходят размеры каких-либо заготовок.

По выбранным картам раскроя (см рис. 7.3) заполняем таблицу:

Листы	Варианты раскроя	Заготовки		Отходы в условных единицах
		А	Б	
1	1	4	0	12
	2	3	4	5
	3	1	9	3
	4	0	12	0

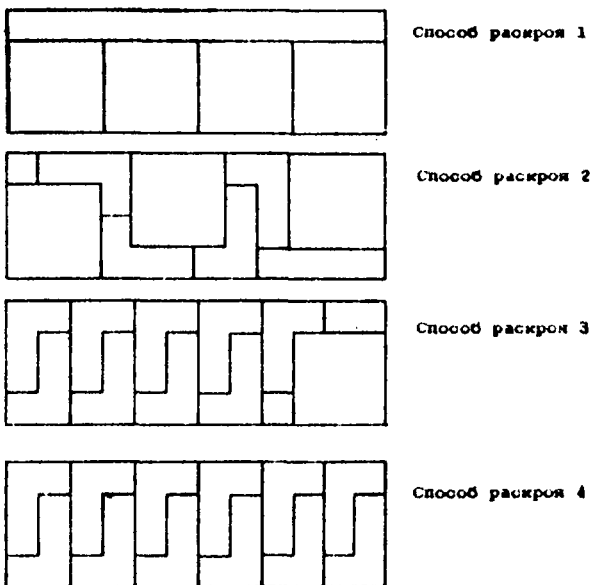


Рис 7.3. Варианты раскроя листа проката

Вводим обозначения:

x_1 - количество листов, раскраиваемых способом 1; x_2 , x_3 , x_4 - способами 2, 3 и 4 соответственно.

Формализуем условия задачи.

$$z = 12x_1 + 5x_2 + 3x_3 \rightarrow \min;$$

$$4x_1 + 3x_2 + x_3 = 180;$$

$$4x_2 + 9x_3 + 12x_4 = 900;$$

$$x_1, x_2, x_3, x_4 \geq 0.$$

Полученную задачу линейного программирования решаем симплекс-методом. Для определения начального допустимого базиса добавляем два фиктивных аргумента x_5 и x_6 :

$$z = 12x_1 + 5x_2 + 3x_3 \rightarrow \min;$$

$$4x_1 + 3x_2 + x_3 + x_5 = 180;$$

$$4x_2 + 9x_3 + 12x_4 + x_6 = 900;$$

$$x_1, x_2, x_3, x_4, x_5, x_6 \geq 0$$

и выбираем их в качестве начального допустимого базиса:

$$z = 12x_1 + 5x_2 + 3x_3 \rightarrow \min;$$

$$x_5 = 180 - 4x_1 - 3x_2 - x_3;$$

$$x_6 = 900 - 4x_2 - 9x_3 - 12x_4;$$

$$x_1, x_2, x_3, x_4, x_5, x_6 \geq 0.$$

Посредством двух итераций симплекс-метода переводим фиктивные переменные из базисных в свободные:

№	Базис	Значение	$-x_1$	$-x_2$	$-x_3$	$-x_4$
1	x_5	180	4	3	1	0
	x_6	900	0	4	9	12
	z	540	3	9	3	0

№	Базис	Значение	$-x_1$	$-x_2$	$-x_3$	$-x_4$
2	x_1	45	0,25	0,75	0,25	0
	x_6	900	0	4	9	12
	z	540	3	4	0	0

№	Базис	Значение	$-x_1$	$-x_2$	$-x_3$	$-x_4$
3	x_1	45	0,25	0,75	0,25	0
	x_4	900/12	0	1/3	9/12	1/12
	z	540	3	4	0	0

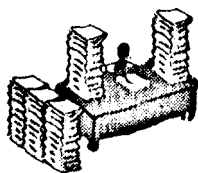
Фиктивные переменные переведены в свободные, исключаем их из задач

№	Базис	Значение	$-x_2$	$-x_3$
4	x_1	45	0,75	0,25
		60	1,333	0,333
	x_4	75	1/3	9/12
		-20	-0,444	-0,111
	z	540	4	0
		-240	-5,333	-1,333

Анализ строки z показывает, что целевая функция может быть уменьшена при переводе переменной x_2 в базис.

№	Базис	Значение	$-x_1$	$-x_3$
4	x_2	60	1,333	0,333
	x_4	55	-0,444	4,388
	z	300	-5,3334	-1,333

Задача решена, для изготовления изделий необходимо 60 листов проката раскроить способом 2 и 55 – способом 4.



Содержание отчета

1. Название работы.
2. Постановка задачи.
3. Эскизы листов проката и заготовок
4. Варианты раскроя листов проката на заготовки
5. Математическая модель задачи
6. Решение задачи симплекс-методом.

Контрольные вопросы

1. Сформулировать общую ЗЛП.
2. Сформулировать основную ЗЛП.
3. Область применения графического метода решения ЗЛП
4. Что такое допустимое решение ЗЛП (допустимый базис)?
5. Понятие базисных и свободных переменных.
6. Как определить количество базисных переменных ЗЛП?
7. Как найти начальный допустимый базис?

Лабораторная работа № 8

МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ ФОРМООБРАЗОВАНИЯ ПОВЕРХНОСТЕЙ ДЕТАЛЕЙ ПРИ ОБРАБОТКЕ НА МЕТАЛЛОРЕЖУЩИХ СТАНКАХ



Цель работы

Овладение навыками разработки математических моделей, алгоритмов, компьютерных программ и численного исследования точности формообразования поверхностей деталей машин и приборов при их обработке на металлорежущих станках.

Основные положения

Формообразующей системой (ФС) станка называется совокупность механических элементов технологической системы, взаимное положение и перемещение которых обеспечивает заданную траекторию движения режущего инструмента относительно обрабатываемой детали.

Звено ФС – это совокупность узлов станка и других элементов технологической системы, смещения которых относительно друг друга в процессе функционирования станка номинально отсутствуют. ФС металлорежущих станков обладает следующей важной особенностью: два последовательных звена имеют в относительном движении не более одной степени свободы. Это означает, что звено может либо поворачиваться относительно оси, фиксированной в соседнем звене, либо двигаться поступательно вдоль фиксированной оси, либо быть относительно неподвижным. Физическая причина такой особенности лежит в необходимости иметь конструктивно неизменную систему опор для перемещения каждого звена, что обеспечивает точность, производительность и надежность станка в целом.

Наиболее простой и одновременно самой распространенной является рассматриваемая в данной работе ФС, содержащая один режущий инструмент, обрабатывающий одну поверхность на одной детали.

Координатный код формообразующей системы. Структура и ряд важных свойств ФС в компактной и однозначной форме могут быть записаны в виде некоторых наборов целых чисел, называемых характеристическими кодами ФС.

При разработке математической модели формообразования используется координатный код ФС, который составляется следующим образом. Все звенья ФС нумеруются, начиная от звена, содержащего заготовку, и кончая звеном, содержащим режущий инструмент. При этом заготовке присваивается номер 0. Далее с каждым звеном i связывается декартова система координат $O_i x_i y_i z_i$, обозначаемая для краткости S_i . Начало O_i и оси систем координат S_i связываются с конструктивными или технологическими базами узла, например, направляющими, шпиндельными опорами и т.п.

Поскольку звенья ФС обладают лишь одной степенью свободы в относительном движении, то с каждым звеном в зависимости от его движения относительно предыдущего звена однозначно связывается один из приведенных в таблице символов:

Наименование оси	Движение звена	
	Поступательное вдоль оси	Вращательное вокруг оси
X	1	4
Y	2	5
Z	3	6

Начальное звено принято считать неподвижным. Число подвижных звеньев обозначим X .

Координатным кодом ФС называется 1-разрядное число $k = k_1 k_2 \dots k_{i-1}$, каждый i -й разряд которого принимает значение от 1 до 6 и отображает в соответствии с таблицей возможное движение i -го звена ФС относительно предыдущего ($i-1$)-го.

Преобразование координат. Разработка математической модели базируется на математическом аппарате преобразования координат, причем в данном случае удобно использовать матрицы четвертого порядка. Благодаря этому любые преобразования координат могут быть выражены с помощью одной математической операции умножения матриц, в то время как при использовании

векторов и матриц третьего порядка необходимы две операции: преобразование поворота системы моделируется умножением матриц, а преобразование сдвига - сложением векторов.

Рассмотрим две системы координат S_{i-1} и S_i . Одна и та же точка пространства имеет в этих системах различные координаты, за исключением тривиального случая, когда S_{i-1} и S_i полностью совпадают.

Пусть $(x_{i-1}, y_{i-1}, z_{i-1})^T$ и $(x_i, y_i, z_i)^T$ - координаты одной и той же точки пространства в системе координат S_{i-1} и S_i . Эти координаты связаны матричным соотношением

$$\begin{pmatrix} x_{i-1} \\ y_{i-1} \\ z_{i-1} \\ 1 \end{pmatrix} = (A_{i-1,i}) \begin{pmatrix} x_i \\ y_i \\ z_i \\ 1 \end{pmatrix}, \quad (8.1)$$

где $A_{i-1,i}$ - матрица порядка 4×4 преобразования координат, имеющая структуру

$$(A) = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad (8.2)$$

причем левый верхний блок 3×3 описывает поворот системы S_i относительно своего начала координат O_i в такое положение, что оси систем S_{i-1} и S_i оказываются параллельными друг другу и одинаково направленными, а три первых элемента a_{14} , a_{24} и a_{34} четвертого столбца матрицы A представляют собой декартовы координаты положения начала координат O_i системы S_i в системе S_{i-1} .

В силу указанного выше свойства ФС рассматриваемых станков системы координат S_{i-1} и S_i в нашем случае совмещаются в результате одного из шести возможных элементарных преобразований - перемещением системы S_{i-1} вдоль одной из осей $O_{i-1}X_{i-1}$, $O_{i-1}Y_{i-1}$ или $O_{i-1}Z_{i-1}$ на величины Δx , Δy или

Δx или поворотом вокруг указанных осей на углы δx , δy или δz . Следовательно, связи между координатами точки устанавливаются следующим образом.

$$\begin{pmatrix} x_{i-1} \\ y_{i-1} \\ z_{i-1} \\ 1 \end{pmatrix} = (A_{i-1,i}^j) \begin{pmatrix} x_i \\ y_i \\ z_i \\ 1 \end{pmatrix}, \quad (8.3)$$

где $A_{i-1,i}^j$, $j = 1, \dots, 6$, - одна из шести матриц, представленных в табл. 8.1 и описывающих указанные элементарные преобразования. Поступательным перемещениям соответствуют матрицы A^1 , A^2 и A^3 , вращательным - матрицы A^4 , A^5 и A^6 .

Таблица 8.1

Матрицы элементарных преобразований координат

Вид движения	Матрица
1	2
Поступательное вдоль оси Ox	$A^1 = \begin{pmatrix} 1 & 0 & 0 & \Delta x \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$
Поступательное вдоль оси Oy	$A^2 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & \Delta y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$
Вращательное вдоль оси Oy	$A^5 = \begin{pmatrix} \cos(\delta y) & 0 & \sin(\delta y) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(\delta y) & 0 & \cos(\delta y) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$
Поступательное вдоль оси Oz	$A^3 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta z \\ 0 & 0 & 0 & 1 \end{pmatrix}$

1	2
Вращательное вокруг оси Ox	$A_6 = \begin{pmatrix} \cos(\delta x) & -\sin(\delta x) & 0 & 0 \\ \sin(\delta x) & \cos(\delta x) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$

Геометрические интерпретации элементарных преобразований координат по формуле (8.3) приведены на рис. 8.1. При этом:

а) линейные смещения Δx , Δy , Δz считаются положительными, если система S_{i-1} движется до совмещения с S_i в положительном направлении соответствующей оси,

б) углы поворота δx , δy , δz считаются положительными, если система S_{i-1} вращается до совмещения с S_i против часовой стрелки (при взгляде от положительного конца оси вращения).

Функция формообразования. Обработанную поверхность детали можно представить как след, оставляемый формообразующими точками инструмента на заготовке. Следовательно, чтобы получить координаты точек обработанной поверхности детали, достаточно установить взаимосвязь между координатами формообразующих точек инструмента в системе координат S_1 , связанной со звеном, содержащим инструмент, и в системе координат S_0 , связанной со звеном, содержащим заготовку. Эта связь устанавливается следующим образом:

$$\begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = (A_{0,1}^{j_1}) \cdot (A_{1,2}^{j_2}) \cdot \dots \cdot (A_{1-1,1}^{j_{1-1}}) \begin{pmatrix} x'' \\ y'' \\ z'' \\ 1 \end{pmatrix}, \quad (8.4)$$

где последовательность j_1, j_2, \dots, j_{1-1} верхних индексов матриц представляет собой координатный код ФС.

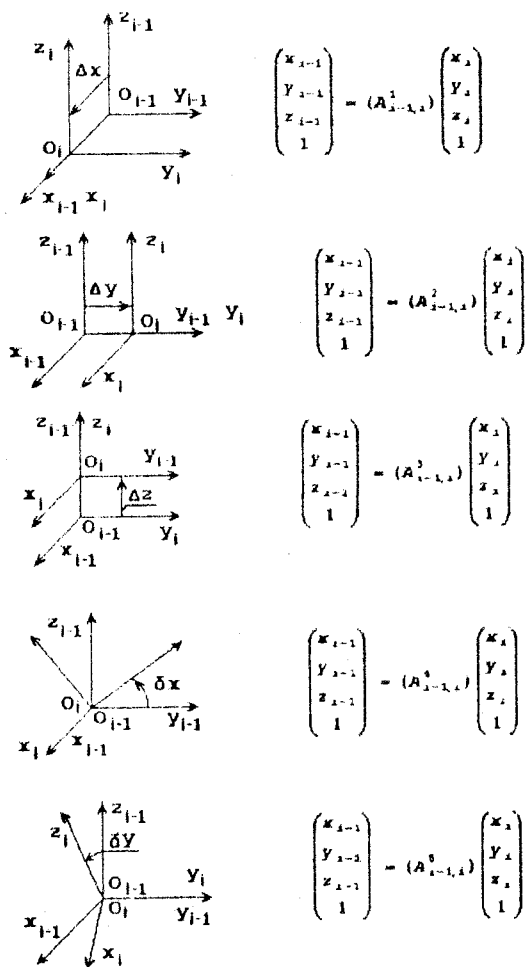


Рис. 8.1. Геометрическая интерпретация элементарных преобразований координат

Выражение (8.4) называется функцией формообразования и является основной математической модели формообразования. Она позволяет получить координаты x , y и z точки на обработанной поверхности детали при заданных относительных положениях звеньев ФС станка, описываемых матрицами $(A_{0,1}^j), (A_{1,2}^j), \dots, (A_{1-1,1}^j)$, и при известной математической модели инструмента $(x'' y'' z'')$.

Модель режущего инструмента. В контексте рассматриваемой задачи под математической моделью инструмента понимается система уравнений и (или) координат его формообразующих элементов. Рассмотрим несколько случаев, отличающихся особенностями составления модели инструмента.

Точечный инструмент. В наиболее простом случае обработанная поверхность может рассматриваться как след одной точки инструмента. Например, при точении цилиндрической поверхности такой точкой является вершина остро заточенного резца. В таких случаях наиболее удобно начало системы координат инструмента поместить в эту точку. Тогда математическая модель инструмента принимает вид

$$\begin{pmatrix} x'' \\ y'' \\ z'' \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}. \quad (8.5)$$

Однолезвийный инструмент. В этом случае множество формообразующих точек инструмента представляют собой линию. Для составления математической модели такого инструмента достаточно записать уравнение этой линии в параметрической форме и установить пределы изменения параметра:

$$\begin{pmatrix} x'' \\ y'' \\ z'' \end{pmatrix} = \begin{pmatrix} x''(\tau) \\ y''(\tau) \\ z''(\tau) \end{pmatrix}, \quad \tau_n \leq \tau \leq \tau_k. \quad (8.6)$$

Здесь τ - параметр, изменяя который от его начального значения τ_n до конечного τ_k , будем получать координаты различных точек лезвия.

Многолезвийный инструмент. Математическая модель многолезвийного инструмента имеет следующий вид:

$$\begin{pmatrix} x_u \\ y_u \\ z_u \end{pmatrix} = \begin{pmatrix} x_u(\tau, a_i) \\ y_u(\tau, a_i) \\ z_u(\tau, a_i) \end{pmatrix}, \quad \tau_n \leq \tau \leq \tau_k, \quad i = \overline{1, n}, \quad (8.7)$$

где n - количество лезвий; a_i - параметр, идентифицирующий i -е лезвие.

Поверхностный инструмент. Математическая модель инструмента, формообразующим элементом которого является поверхность, имеет следующий вид:

$$\begin{pmatrix} x_u \\ y_u \\ z_u \end{pmatrix} = \begin{pmatrix} x_u(u, v) \\ y_u(u, v) \\ z_u(u, v) \end{pmatrix}, \quad u_n \leq u \leq u_k, \quad v_n \leq v \leq v_k, \quad (8.8)$$

где u и v - свободные параметры; u_n, u_k, v_n, v_k - пределы, в которых они принимают значения.

Постановка задачи и варианты заданий

Разработать имитационную математическую модель, алгоритм, компьютерную программу и выполнить численные исследования формообразования поверхности 1. Эскиз детали, инструмента и схема станка представлены на одном из рис. 8.4 – 8.9 в соответствии с вариантом задания.

Методические указания



Порядок выполнения работы проиллюстрируем на примере разработки математической модели формообразования цилиндрической поверхности на токарном станке (рис. 8.2).

1. *Выбор систем координат, связанных со звеньями формообразующей системы станка.* В качестве примера на рис. 8.2 показан станок, со звеньями которого связаны системы координат. При выборе последних следует исходить из того, что два последовательных звена в относительном движении имеют не более одной степени свободы, то есть совмещение систем координат соседних звеньев должно осуществляться только одним из рассмотренных выше элементарных движений.

2. *Составление координатного кода* формообразующей системы станка. Примем в качестве нулевого звена ($i=0$) обрабатываемую деталь со шпинделем. Следующее звено ($i=1$) – станина. Считая нулевое звено неподвижным, получим, что в относительном движении станина вращается вокруг оси z_0 в направлении, обратном направлению вращения шпинделя. Этому движению соответствует символ 6, то есть $k_1 = 6$. Аналогично поступательное движение продольного суппорта относительно станины совершается вдоль оси z_1 , то есть $k_2 = 3$, поперечный же суппорт поступательно движется вдоль оси x_2 , то есть $k_3 = 1$. Таким образом, для токарно-винторезного станка координатный код ФС представляет собой трехзначное число $k=631$.

3. *Построение математической модели инструмента.* Для более полного освещения математического моделирования возможных видов инструмента рассмотрим несколько случаев, отличающихся особенностями составления модели инструмента.

3.1. *Точечный инструмент.* Начало системы координат следует совместить с формообразующей точкой инструмента. Математическая модель принимает вид (8.5).

3.2. *Однолезвийный инструмент.* На рис. 8.3,а показан резец. Система координат S_1 этого инструмента выбрана таким образом, что лезвие AB резца располагается в плоскости Oxz и составляет заданный угол λ с осью O_1z_1 . Выражение (8.6) для рассматриваемого случая однолезвийного инструмента примет вид

$$\begin{pmatrix} x'' \\ y'' \\ z'' \end{pmatrix} = \begin{pmatrix} S \sin \lambda \\ 0 \\ -S \cos \lambda \end{pmatrix}, \quad 0 \leq S \leq S_1, \quad (8.9)$$

где S_1 - длина лезвия инструмента. Физически S является координатой, отсчитываемой вдоль лезвия инструмента.

3.3. *Многолезвийный инструмент.* На рис. 8.3,б показана цилиндрическая фреза с винтовым зубом и связанная с ней система координат. Математическая модель такого инструмента имеет вид

$$\begin{pmatrix} x'' \\ y'' \\ z'' \end{pmatrix} = \begin{pmatrix} R \cos(\varphi_1 + z/P) \\ R \sin(\varphi_1 + z/P) \\ z \end{pmatrix}, \quad (8.10)$$

где $i = \overline{1, n}$ – номер лезвия; n – число лезвий; φ_i – угол, определяющий положение i -го лезвия на цилиндрической поверхности фрезы, отсчитываемый от оси O_1x_1 против часовой стрелки; R – радиус фрезы, x – независимая переменная, имеющая смысл линейного смещения вдоль оси O_1x_1 ($0 < x < L$); p – параметр, связанный с шагом H винтовой линии лезвия соотношением $H = 2\pi p$

3.4. *Поверхностный инструмент.* Например, математическая модель шлифовального круга, формообразующим элементом которого является цилиндрическая поверхность, имеет следующий вид:

$$\begin{pmatrix} x_{\text{н}}(u, v) \\ y_{\text{н}}(u, v) \\ z_{\text{н}}(u, v) \end{pmatrix} = \begin{pmatrix} R \cos u \\ R \sin u \\ v \end{pmatrix}, \quad 0 \leq u \leq 2\pi, \quad 0 \leq v \leq b, \quad (8.11)$$

где R – радиус цилиндра; b – толщина (или высота) инструмента, u – угол в его сечении, отсчитываемый от некоторой фиксированной оси; v – линейное смещение вдоль образующей.

4. *Составление уравнения формообразования.* Для токарного станка выше нами получен следующий координатный код: $k=631$. Учитывая математическую модель инструмента (8.5) и формулу (8.4), определяющую порядок составления уравнения формообразования, получим уравнение математической модели ФС токарно-винторезного станка

$$\begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = (A_{0,1}^6) (A_{1,2}^3) (A_{2,3}^1) \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}, \quad (8.12)$$

где x, y, z – координаты обработанной точки в системе координат детали.

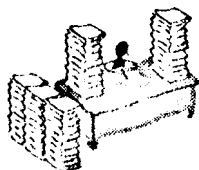
Подставив значения матриц A^6, A^3 и A^1 из табл. 8.1, получим:

$$\begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} \cos \varphi & -\sin \varphi & 0 & 0 \\ \sin \varphi & \cos \varphi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta x \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & \Delta x \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}, \quad (8.13)$$

где φ – угол поворота шпинделя; Δx – смещение системы координат продольного суппорта относительно системы координат станины; Δz – смещение системы координат поперечного суппорта относительно системы координат продольного.

5. *Разработка алгоритма* расчета координат точек обработанной поверхности.

6. *Разработка компьютерной программы и численные исследования.*



Содержание отчета

1. Название работы.
2. Содержание задания (включая эскиз формообразующей системы станка, детали и инструмента).
3. Координатный код формообразующей системы.
4. Математическая модель инструмента.
5. Математическая модель формообразования.
6. Алгоритм расчета координат точек обработанной поверхности.
7. Компьютерная программа и результаты численных исследований.

Контрольные вопросы

1. Дать определение формообразующей системы станка и звена ФС.
2. В какой последовательности составляется координатный код ФС?
3. Какова особенность использования матриц четвертого порядка для преобразования координат?
4. Изложить правила знаков для аргументов матриц преобразования координат?
5. Записать математическую модель формообразования в общем виде.
6. Записать математическую модель инструмента в общем виде.

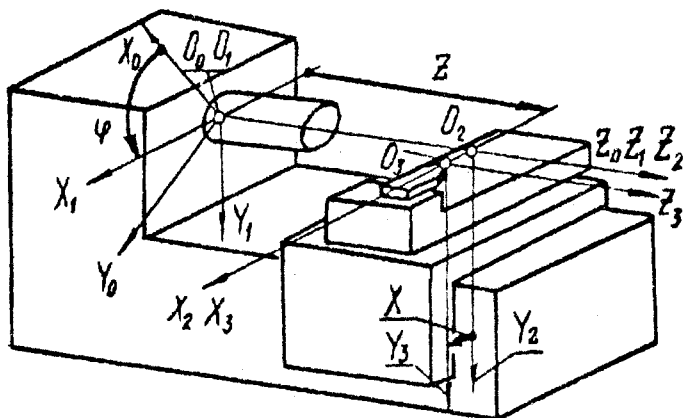


Рис. 8.2. Формообразующая система токарного станка

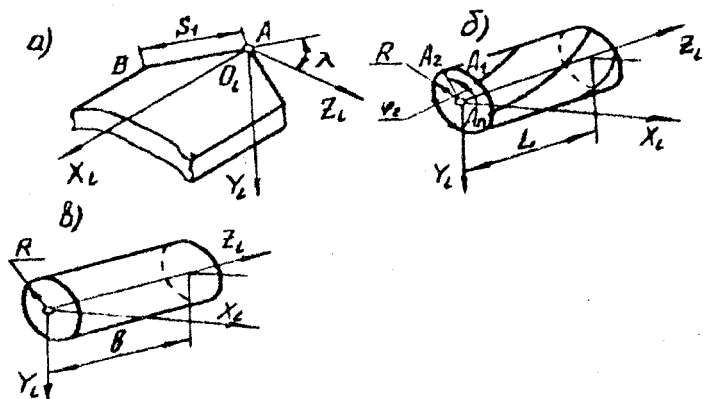


Рис. 8.3. Геометрические модели инструментов:

- а) однолезвийного;
- б) многолезвийного;
- в) поверхностного

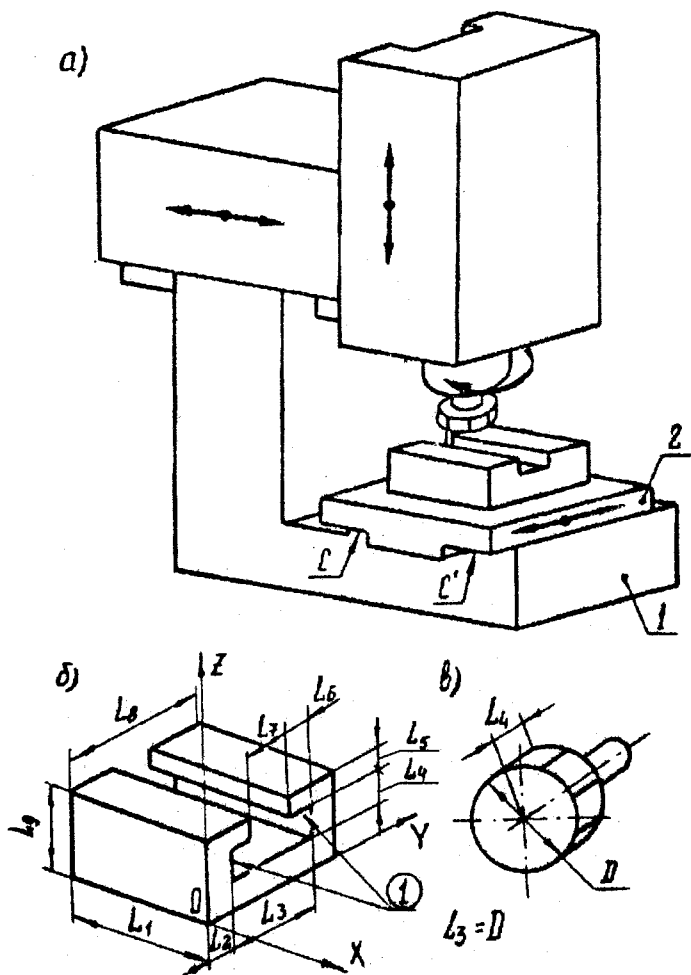


Рис. 8.4. Фрезеровать паз ①
 (вариант 1)

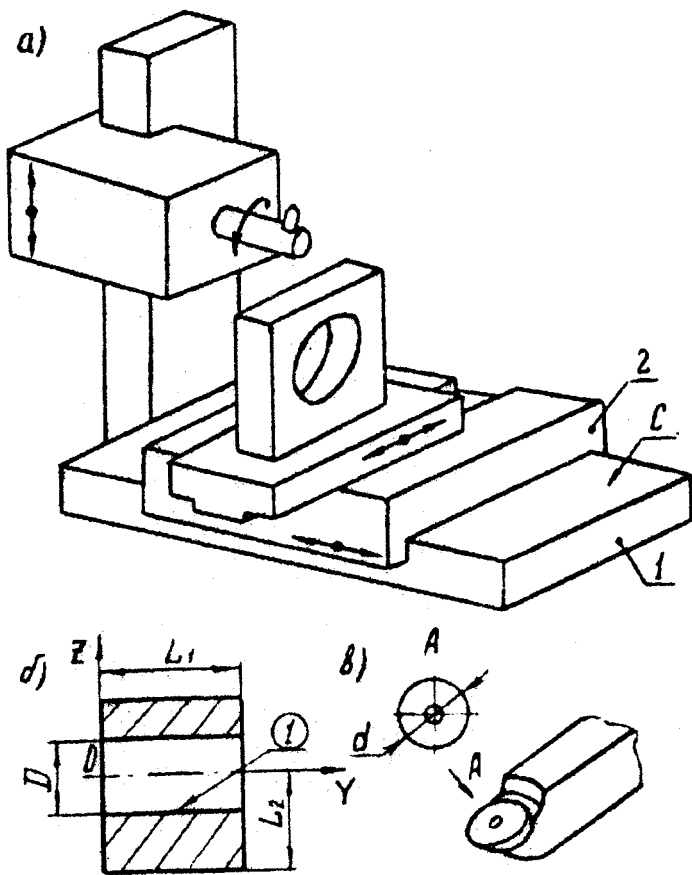


Рис. 8.5. Расточить отверстие (вариант 2) ①

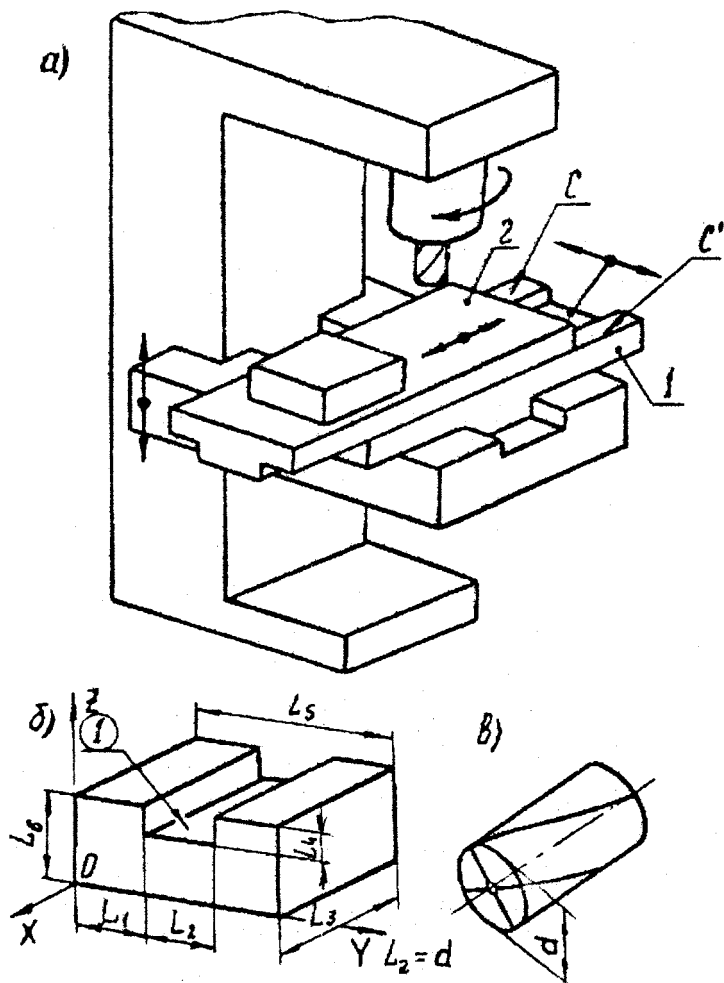


Рис. 8.6. Фрезеровать паз (вариант 3) ①

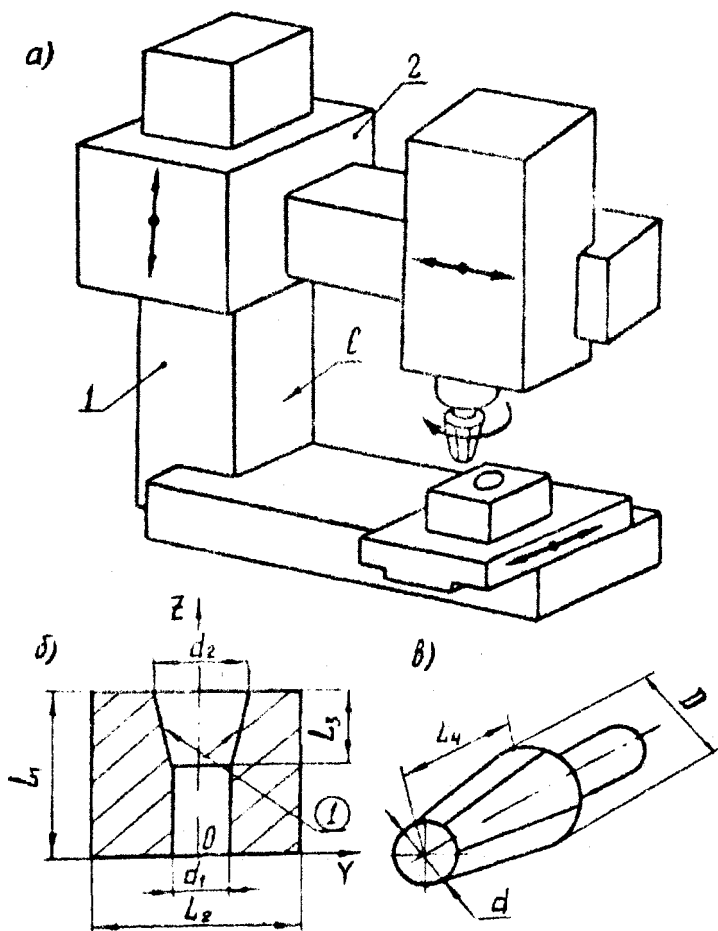


Рис. 8.7. Зенкеровать отверстие ①
 (вариант 4)

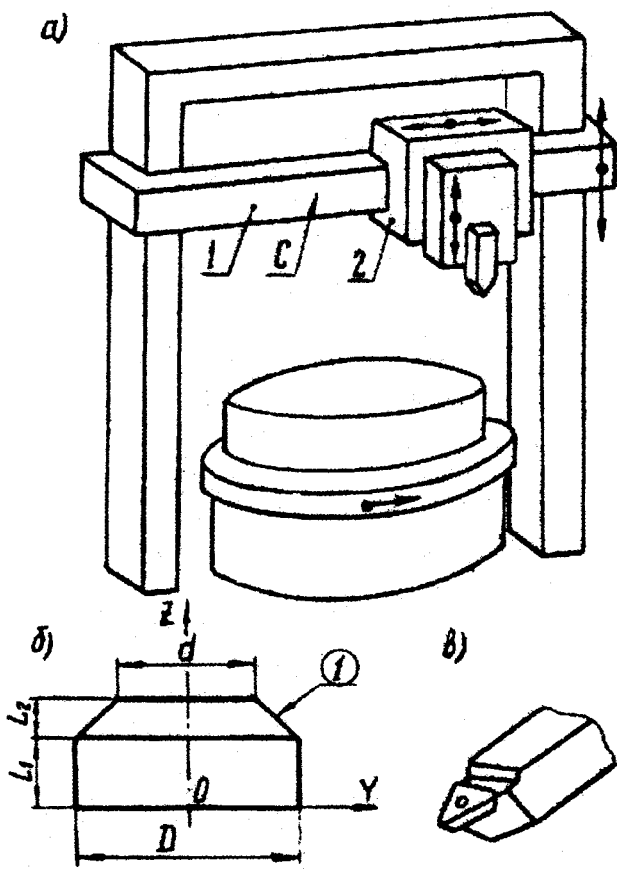


Рис. 8.8. Точить конус ①
(вариант 5)

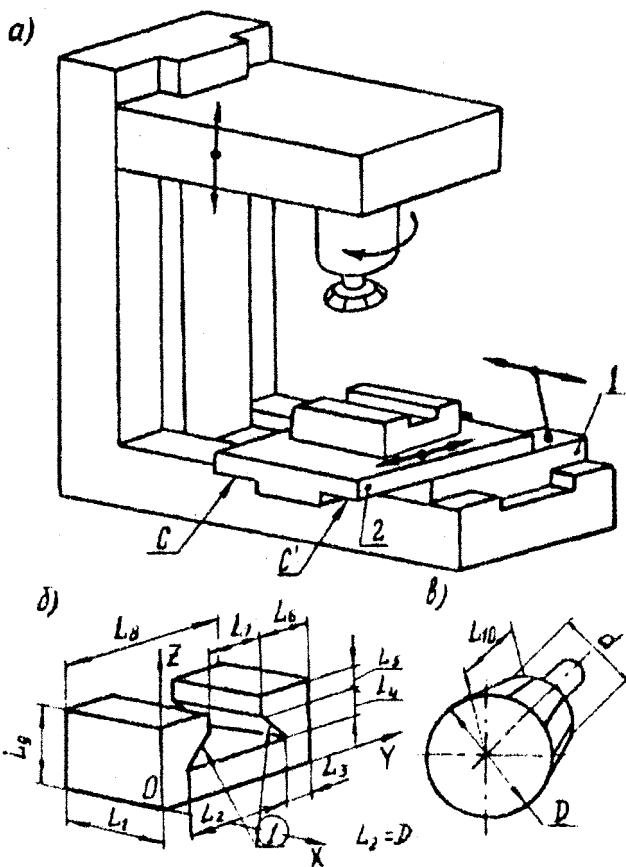


Рис. 8.9. Фрезеровать конический паз ①
(вариант б)

3. НЕКОТОРЫЕ ПОЛОЖЕНИЯ ДИСКРЕТНОЙ МАТЕМАТИКИ

3.1. Некоторые понятия теории множеств

Понятие множества является фундаментальным неопределяемым понятием, как и понятие точки или линии. Интуитивно под *множествам* понимают совокупность определенных вполне различаемых объектов, рассматриваемых по некоторому признаку как единое целое. Отдельные объекты, из которых состоит множество, называются *элементами множества*.

Множество, не содержащее ни одного элемента, называется *пустым*.

Если любой элемент x , принадлежащий множеству X , принадлежит и множеству Y , то множество X называется *подмножеством* множества Y .

Множества, элементами которых являются числа, называются *числовыми*.

Пусть s - числовое множество. *Верхней границей* множества s называется число c , которое не меньше любого его элемента. Точной верхней границей или *супремумом* множества s называют такую его верхнюю границу, которая не больше любой другой. *Нижней границей* множества s называется число c , которое не больше любого его элемента. Точной нижней границей или *инфимумом* множества s называют такую его нижнюю границу, которая не меньше любой другой.

3.2. Основные понятия теории графов

Графом G называется пара (X, U) , где $X = \{x_1, x_2, \dots, x_n\}$ - непустое множество вершин, а $U = \{u_1, u_2, \dots, u_m\}$ - множество отрезков, соединяющих их некоторым образом. Отрезок, соединяющий вершины x_i и x_j , обозначают часто в виде пары $u_{i,j} = (x_i, x_j)$. Если порядок расположения вершин в паре не имеет значения, то отрезок называют *ребром*, а вершины x_i и x_j - *концами* ребра. Если порядок расположения вершин в указанной паре несет смысловую нагрузку и не может быть изменен, то отрезок, их соединяющий, называется *ориентированным ребром* или *дугой*, вершина x_i - *началом*, а x_j - *концом* дуги.

Две вершины x и y называются *смежными*, если они различны и существует дуга или ребро, их соединяющее. Два ребра (две дуги, дуга и ребро) называются *смежными*, если они имеют общую вершину. Дуга u называется *инцидентной* вершине x , если она заходит в эту вершину или исходит из нее. Ребро u инцидентно вершине x , если она является концом этого ребра.

Подграфом G_A графа $G = (X, Y)$ называется граф, в который входит лишь часть вершин графа G вместе с соединяющими их дугами. Если любые две вершины графа соединены ребром или дугой, то граф называется **полным**.

Путь в орграфе G называется такая последовательность дуг $\mu = (u_1, u_2, \dots, u_n)$, в которой конец каждой предыдущей дуги совпадает с началом последующей. Путь, у которого начальная вершина первой дуги совпадает с конечной вершиной последней, называется **контуром**. **Гамильтоновым** называется контур, проходящий через все вершины графа, причем только по одному разу.

Граф называется **взвешенным** или **нагруженным**, если каждому его ребру (дуге) поставлено в соответствие число, которое называется **весом** или **длиной** ребра (дуги). **Весом графа** называется сумма весов составляющих его ребер и дуг, **тонкостью** - вес самого тяжелого, **шириной** - самого легкого из его ребер и дуг.

Длиной пути μ невзвешенного графа называется число $L(\mu) = n$, равное числу дуг, составляющих этот путь. Длина пути взвешенного графа определяется как сумма длин дуг, составляющих этот путь, т.е. $L(\mu) = \sum_{i=1}^n l(u_i)$, где $l(u_i)$ - длина дуги u_i . **Тонкостью** пути называется вес самого тяжелого, **шириной** - самого легкого из его ребер и дуг.

Неориентированный ациклический (т.е. не содержащий циклов) граф называется **деревом**. Если в дереве выделена некоторая вершина, называемая **корнем**, то дерево называется **корневым**.

Неориентированный граф называется **связным**, если любые его две вершины соединены цепью. Несвязный граф состоит из отдельных частей, которые называются его **связными компонентами**. Чтобы определить связность смешанного или орграфа, нужно абстрагироваться от ориентации дуг.

Дерево, которое является подграфом связного графа и содержит все его вершины, называется **остовным** или просто **остовом** этого графа.

3.3. Алгоритм Дейкстры поиска кратчайшего пути

Шаг 1. Начальной вершине x_0 присваиваем метку $d(x_0) = 0$, остальным вершинам x_1 - метку $d(x_1) = +\infty$. Начальную вершину окрашиваем.

Шаг 2. Для каждой неокрашенной вершины x_1 , связанной с последней окрашенной вершиной y дугой (y, x_1) , пересчитываем ее метку по формуле

$$d(x_1) = \min(d(x_1), d(y) + l(y, x_1)),$$

где $l(y, x_1)$ - длина дуги (y, x_1) .

Шаг 3. Среди неокрашенных находим вершину x_s с минимальной меткой и окрашиваем ее и ведущую в нее из одной из ранее окрашенных вершин y_r дугу (y_r, x_s) , удовлетворяющую условию $d(x_s) = d(y_r) + l(y_r, x_s)$. Если при этом окрашенной оказалась конечная вершина, то задача решена, иначе переходим к шагу 2.

По окончании выполнения алгоритма окрашенные вершины и дуги составят корневое дерево с корнем x_0 , отображающее кратчайшие пути из этой вершины во все остальные.

3.4. Алгоритм поиска тончайшего пути

Тончайшим путем между двумя вершинами графа называется путь минимальной тонкости.

Шаг 1. Начальной вершине x_0 присваиваем метку $d(x_0) = -\infty$, остальным вершинам x_i - метку $d(x_i) = +\infty$. Начальную вершину окрашиваем.

Шаг 2. Для каждой неокрашенной вершины x_i , связанной с последней окрашенной вершиной y дугой (y, x_i) , пересчитываем ее метку по формуле

$$d(x_i) = \min(\max(d(y), l(y, x_i)), d(x_i)).$$

Шаг 3. Среди неокрашенных находим вершину x_s с минимальной меткой и окрашиваем ее и ведущую в нее из одной из ранее окрашенных вершин y_r дугу (y_r, x_s) , удовлетворяющую условию $d(x_s) = \max(d(y_r), l(y_r, x_s))$. Если при этом окрашенной оказалась конечная вершина, то задача решена, тонкость искомого пути равна метке конечной вершины, а сам путь отображен на окрашенном дереве. В противном случае переходим к шагу 2.

По окончании выполнения алгоритма окрашенные вершины и дуги составят корневое дерево с корнем x_0 , отображающее тончайшие пути из этой вершины во все остальные.

Путь из вершины s в вершину t графа, длина которого является максимальной, называется **длиннейшим**.

3.5. Алгоритм поиска длиннейшего пути

Шаг 1. Начальной вершине x_0 присваиваем метку $d(x_0) = 0$, остальным вершинам x_1 - метку $d(x_1) = -\infty$. Начальную вершину окрашиваем.

Шаг 2. Для каждой вершины x_1 , связанной с последней окрашенной вершиной y дугой (y, x_1) , пересчитываем ее метку по формуле

$$d(x_1) = \max\{d(x_1), d(y) + l(y, x_1)\},$$

где $l(y, x_1)$ - длина дуги (y, x_1) . Если при этом изменяется метка y ранее окрашенной вершины, то краска с неё и инцидентной ей дуги (ребра) снимается. Если ни одна из меток не изменилась и все вершины уже являются окрашенными, то задача решена, иначе переходим к шагу 3.

Шаг 3. Среди неокрашенных находим вершину x , с максимальной меткой и окрашиваем ее и ведущую в нее из одной из ранее окрашенных вершин y , дугу (y, x) , удовлетворяющую условию $d(x) = d(y) + l(y, x)$. Переходим к шагу 2.

По окончании выполнения алгоритма окрашенные вершины и дуги составят корневое дерево с корнем x_0 , отображающее длиннейшие пути из этой вершины во все остальные.

3.6. Метод ветвей и границ

Многие прикладные задачи сводятся к известной задаче о коммивояжере, которая формулируется следующим образом.

Имеется n городов. Известны расстояния a_{ij} между ними $\forall i, j = \overline{1, n}$. Если прямого маршрута между некоторыми из них не существует, то соответствующее расстояние между ними принимается равным бесконечности. Коммивояжер, отправляясь из какого-либо города, должен посетить все остальные, побывав в каждом из них только по одному разу, и вернуться назад. Необходимо определить такую последовательность посещения городов, при которой длина маршрута будет минимальной.

Если городам поставить в соответствие вершины взвешенного графа, а дорогам - ребра или дуги (если движение по каким-либо маршрутам возможно только в одном направлении), нагруженные расстояниями между городами, то задача сводится к поиску гамильтонового контура минимальной длины.

Для решения задачи о коммивояжере разработано достаточно большое число алгоритмов. Одним из наиболее эффективных считается метод ветвей и

грани, разработанный в 1963 году Дж. Литтлом, К. Мурти, Д. Сунни и К. Кэролом. Сущность этого метода заключается в следующем.

Множество всевозможных гамильтоновых контуров обозначим Ω^0 . Пусть $l(\mu)$ - длина контура $\mu \in \Omega^0$. Предположим, есть правило, которое позволяет найти нижнюю границу ϕ^0 множества длин этих контуров. Таким образом, $\forall \mu \in \Omega^0, l(\mu) \geq \phi^0$. Затем множество Ω^0 всех гамильтоновых контуров разбивается на два подмножества Ω_{ij}^1 и $\overline{\Omega}_{ij}^1$, первое из которых состоит из всех тех и только тех из них, которые включают в себя дугу (x_i, x_j) , другое - из гамильтоновых контуров, не содержащих этой дуги. Дуга (x_i, x_j) , которая в данном случае называется разбивающей дугой, выбирается также по определенному правилу. Для каждого из полученных подмножеств Ω_{ij}^1 и $\overline{\Omega}_{ij}^1$ находим нижние границы ϕ_{ij}^1 и $\overline{\phi}_{ij}^1$ множеств длин принадлежащих им контуров. Так как $\Omega_{ij}^1 \subset \Omega^0$, то в силу известной из теории множеств теоремы $\phi_{ij}^1 \geq \phi^0$, так как $\overline{\Omega}_{ij}^1 \subset \Omega^0$, то $\overline{\phi}_{ij}^1 \geq \phi^0$. Сравниваем две полученные нижние границы ϕ_{ij}^1 и $\overline{\phi}_{ij}^1$. Если $\phi_{ij}^1 < \overline{\phi}_{ij}^1$, то более вероятно, что гамильтонов контур минимальной длины находится во множестве Ω_{ij}^1 , и наоборот, если $\phi_{ij}^1 > \overline{\phi}_{ij}^1$, то более вероятно, что искомым контур находится во множестве $\overline{\Omega}_{ij}^1$. То подмножество, которое с большей вероятностью содержит гамильтонов контур минимальной длины, аналогичным образом разбиваем на два подмножества Ω_{kl}^2 и $\overline{\Omega}_{kl}^2$, первое из которых состоит из гамильтоновых контуров, включающих в себя дугу (x_k, x_l) , второе - не включающих. Далее определяются нижние границы ϕ_{kl}^2 и $\overline{\phi}_{kl}^2$ этих подмножеств, которые сравниваются между собой, и т.д., пока в результате дробления подмножеств не получим подмножество, содержащее единственный гамильтонов контур. Эту процедуру удобно представлять в виде корневого дерева, как это показано на рис. 3.1.

Нижней границей для подмножества, содержащего единственный контур, является длина этого контура. Эту величину называют рекордом и обозначают Rec . Получив гамильтонов контур, просматривают оборванные ветви дерева и сравнивают нижние границы для подмножеств, соответствующих этим ветвям, с рекордом. Если все указанные границы больше рекорда, то найденный контур имеет минимальную длину, следовательно, задача решена. Дуги, составляющие этот контур, устанавливаются путем просмотра вершин пути, ведущего из кор-

ния построенного дерева в соответствующий ему лист. Если какая-либо из границ оказывается меньше рекорда, то вполне вероятно, что соответствующее подмножество содержит контур, длина которого меньше рекорда. В этом случае такие подмножества дробятся в соответствии с рассмотренной выше процедурой. В результате развития оборванных ветвей могут быть получены новые гамильтоновы контуры. В этом случае в качестве рекорда выбирается наименьшая из их длин. Процесс дробления подмножеств продолжается до тех пор, пока все нижние границы не станут больше рекорда. Решением задачи является гамильтонов контур, длина которого равна рекорду.

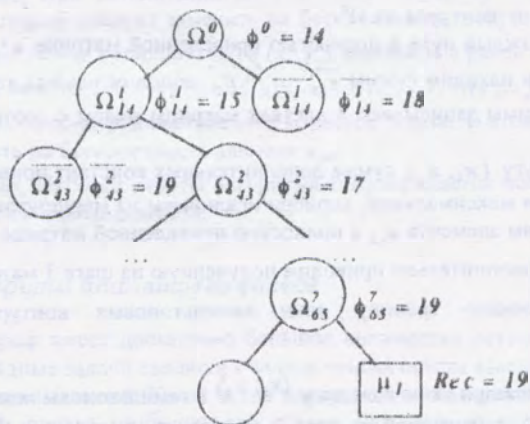


Рис. 3.1 Дерево решения задачи о коммивояжере

3.7. Алгоритм Литтла решения задачи о коммивояжере

В приведенном ниже алгоритме использованы следующие понятия.

Приведение матрицы по строкам заключается в поиске в каждой строке i матрицы расстояний A минимального элемента $a_{i1} = \min_j(a_{ij})$ и вычитании его из всех элементов данной строки. В результате каждая строка будет содержать по крайней мере один нулевой элемент. **Приведение матрицы по столбцам** заключается в поиске в каждом столбце j матрицы расстояний A мини-

мального элемента $\beta_j = \min_i \{a_{ij}\}$ и вычитании его из всех элементов этого столбца. Матрицу, приведенную как по строкам, так и по столбцам, называют *полностью приведенной*, а величины α_i и β_j ($i, j = \overline{1, n}$) - *константами приведения*.

Шаг 1. Приводим матрицу расстояний по строкам и столбцам. Находим нижнюю границу

$$\phi^0 = \gamma = \sum_{i=1}^n \alpha_i + \sum_{j=1}^n \beta_j$$

множества длин контуров из Ω^0 .

Шаг 2. Каждый нуль в полностью приведенной матрице a^0 мысленно заменяем на ∞ и находим суммы $\overline{\gamma}_{ij}' = \alpha_i' + \beta_j'$ дополнительных констант приведения. Эти суммы записываем в клетках матрицы рядом с соответствующими им нулями.

Шаг 3. Дугу (x_i, x_j) , сумма дополнительных констант приведения для которой является максимальной, априори исключаем из маршрутов коммивояжера путем замены элемента a_{ij} в полностью приведенной матрице a^0 на ∞ .

Шаг 4. Дополнительно приводим полученную на шаге 3 матрицу \overline{a}_{ij}^1 и определяем нижнюю границу длин гамильтоновых контуров $\mu \in \overline{\Omega}_{ij}^1$: $\overline{\phi}_{ij}^1 = \gamma + \overline{\gamma}_{ij}'$.

Шаг 5. Априори включаем дугу (x_i, x_j) в гамильтоновы контуры, для чего из матрицы a^0 , полученной на шаге 1, удаляем i -ю строку и j -й столбец, а элемент a_{ij} заменяем на ∞ .

Шаг 6. Приводим матрицу \overline{a}_{ij}^1 , полученную на шаге 5, и определяем нижнюю границу длин контуров из $\overline{\Omega}_{ij}^1$: $\phi_{ij}^1 = \gamma + \overline{\gamma}_{ij}'$, где $\overline{\gamma}_{ij}'$ - сумма констант приведения матрицы.

Шаг 7. Проверяем размерность сокращенной матрицы \overline{a}_{ij}^1 . Если ее размерность 2×2 , то переходим на шаг 9, в противном случае выполняем шаг 8.

Шаг 8. Сравниваем нижние границы ϕ_{ij}^1 и $\overline{\phi}_{ij}^1$ и переходим к шагу 2. Если при этом $\overline{\phi}_{ij}^1 \leq \phi_{ij}^1$, то дальнейшему разбиению подлежит множество $\overline{\Omega}_{ij}^1$, т.е. на шаге 2 анализируется матрица \overline{a}_{ij}^1 , полученная на шаге 4; если же $\phi_{ij}^1 \leq \overline{\phi}_{ij}^1$, то дальнейшему разбиению подлежит множество Ω_{ij}^1 и на шаге 2 анализируется матрица a_{ij}^1 , полученная на шаге 6.

Шаг 9. Определяем найденный гамильтонов контур и его длину Rec .

Шаг 10. Сравниваем Rec с нижними границами оборванных ветвей. Если Rec не превышает указанных нижних границ, то задача решена, контур длины Rec является кратчайшим. Если же длина найденного контура больше некоторых нижних границ, то, действуя в соответствии с приведенным алгоритмом, развиваем эти ветви до тех пор, пока все нижние границы оборванных ветвей не станут больше Rec . Если в результате развития оборванных ветвей будут получены новые гамильтоновы контуры, то в качестве рекорда выбирается наименьшая из их длин.

Примечание. При выполнении шага 5 может возникнуть ситуация, что элемент a_{ij} , который следует заменить на бесконечность, отсутствует. Это означает, что включаемая в маршрут дуга (x_i, x_j) составила с ранее включенными в него дугами некоторый путь $((x_i, x_j), (x_j, x_k), \dots, (x_k, x_l), \dots, (x_l, x_m))$. Этот путь нетрудно найти, просмотрев построенное корневое дерево. В этом случае необходимо заменить на бесконечность элемент a_{ij} .

Примечание. Процесс работы алгоритма сопровождается построением иллюстрирующего корневого дерева.

3.8. Алгоритм ближайшего соседа

Обычно граф имеет достаточно большое количество остовных деревьев. Многие прикладные задачи сводятся к задаче поиска остова взвешенного графа, вес которого минимальный. Такой остов называется минимальным. Ниже приведен алгоритм поиска минимального остова, называемый алгоритмом ближайшего соседа.

Шаг 1. Определяем число вершин $|X|$ графа $G = (X, Y)$, число найденных ребер искомого минимального остова $k := 0$, выбираем и окрашиваем произвольную вершину x .

Шаг 2. Рассматриваем все ребра, у которых один конец является окрашенной, другой неокрашенной вершиной, окрашиваем и включаем в дерево легчайшее из них, увеличиваем $k := k + 1$, окрашиваем инцидентную этому ребру вторую вершину. Если $k = |X| - 1$, то задача решена и окрашенные ребра составляют искомым минимальный остов, иначе повторяем шаг 2.

4. ОСНОВНЫЕ ПОЛОЖЕНИЯ ТЕОРИИ ЛИНЕЙНОГО ПРОГРАММИРОВАНИЯ

4.1. Решение общей задачи линейного программирования

Общей задачей линейного программирования называется задача поиска экстремума (максимума или минимума) линейной функции n переменных x_1, x_2, \dots, x_n

$$z = c_1 x_1 + c_2 x_2 + \dots + c_n x_n \rightarrow \min(\max),$$

удовлетворяющих m линейным ограничениям

$$a_{i1} x_1 + a_{i2} x_2 + \dots + a_{in} x_n \leq b_i, \quad i = \overline{1, m_1};$$

$$a_{i1} x_1 + a_{i2} x_2 + \dots + a_{in} x_n = b_i, \quad i = \overline{m_1 + 1, m_2};$$

$$a_{i1} x_1 + a_{i2} x_2 + \dots + a_{in} x_n \geq b_i, \quad i = \overline{m_2 + 1, m}$$

и, возможно, условию неотрицательности для некоторых переменных.

Основной задачей линейного программирования называется задача поиска минимума линейной функции n переменных x_1, x_2, \dots, x_n

$$z = c_1 x_1 + c_2 x_2 + \dots + c_n x_n \rightarrow \min(\max),$$

удовлетворяющих m линейным ограничениям

$$a_{11} x_1 + a_{12} x_2 + \dots + a_{1n} x_n = b_1;$$

$$a_{21} x_1 + a_{22} x_2 + \dots + a_{2n} x_n = b_2;$$

...

$$a_{m1} x_1 + a_{m2} x_2 + \dots + a_{mn} x_n = b_m$$

и условию неотрицательности

$$x_1, x_2, \dots, x_n \geq 0.$$

Если общая задача линейного программирования зависит только от двух переменных x_1 и x_2 , то ее следует решать графическим методом.

Если же переменных больше двух, следует поступать следующим образом:

□ общую задачу линейного программирования свести к основной;

- полученную основную задачу линейного программирования решить симплекс-методом;
- результат решения основной задачи линейного программирования трансформировать в результат исходной общей задачи.

Чтобы общую задачу линейного программирования свести к основной, следует выполнить следующие действия:

- если в общей задаче линейного программирования требуется найти максимум целевой функции, то последняя умножается на «-1» и ищется минимум полученной таким образом новой функции;
- если общая задача линейного программирования содержит ограничение вида $a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n \leq b_1$, то вводится фиктивная переменная $x_{n+1} \geq 0$ и ограничение приводится к виду $a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n + x_{n+1} = b_1$;
- если общая задача линейного программирования содержит ограничение вида $a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n \geq b_1$, то вводится фиктивная переменная $x_{n+1} \geq 0$ и ограничение приводится к виду $a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n - x_{n+1} = b_1$;
- если переменная x_j общей задачи линейного программирования может принимать как положительные, так и отрицательные значения, то вводятся две фиктивные переменные $x_{n+1} \geq 0$ и $x_{n+2} \geq 0$, и переменная x_j заменяется их разностью $x_j = x_{n+1} - x_{n+2}$.

Чтобы получить решение общей задачи линейного программирования из решения соответствующей ей основной, необходимо воспользоваться следующей теоремой линейного программирования.

Теорема. Если общая задача линейного программирования, зависящая от переменных x_1, x_2, \dots, x_n , была сведена к основной, зависящей от переменных $x_1, x_2, \dots, x_n, x_{n+1}, \dots, x_{n+k}$, и величины $x_1^*, x_2^*, \dots, x_n^*, x_{n+1}^*, \dots, x_{n+k}^*$ являются решением последней, то $x_1^*, x_2^*, \dots, x_n^*$ - решение исходной общей задачи.

4.2. Некоторые положения теории линейного программирования

Основная задача линейного программирования может быть записана в матричной форме

$$z = C^T X \rightarrow \min,$$

$$AX = B,$$

$$X \geq 0,$$

$$\text{где } C = \begin{pmatrix} c_1 \\ c_2 \\ \dots \\ c_n \end{pmatrix}, X = \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{pmatrix}, B = \begin{pmatrix} b_1 \\ b_2 \\ \dots \\ b_m \end{pmatrix}, A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{pmatrix}.$$

Рассмотрим множество U точек X , удовлетворяющих условиям

$$U = \{X \in E^n / AX = B, X \geq 0\}.$$

Это множество называется множеством допустимых решений. В теории линейного программирования установлены следующие его свойства, на которых основывается симплекс-метод решения задачи.

- Множество U , если оно не пусто, является выпуклым многогранником в пространстве E^n .
- Целевая функция принимает оптимальное значение в одной из угловых точек (вершин) многогранника решений.
- Если целевая функция принимает оптимальное решение более чем в одной вершине многогранника решений, то она принимает это же значение во всех точках, являющихся выпуклыми линейными комбинациями данных вершин.

Из приведенных свойств видно, что решение основной задачи линейного программирования следует искать среди угловых точек многогранника решений. Отличительный признак угловых точек приведен в следующей ниже теореме.

Столбцы матрицы ограничений A обозначим через A_1, A_2, \dots, A_n . Тогда система ограничений может быть записана в виде

$$A_1 x_1 + A_2 x_2 + \dots + A_n x_n = B.$$

Теорема. Пусть U - множество допустимых решений, $A \neq 0$, $r = \text{rang } A$. Для того, чтобы точка $X^* = (x_1^*, x_2^*, \dots, x_n^*) \in U$ была угловой точкой множества U , необходимо и достаточно, чтобы существовали номера j_1, j_2, \dots, j_r столбцов матрицы A такие, что

$$A_{j_1} x_{j_1}^* + A_{j_2} x_{j_2}^* + \dots + A_{j_r} x_{j_r}^* = B;$$

для всех остальных номеров $j \neq j_1, j_2, \dots, j_r$.

$$x_j^* = 0,$$

причем указанные столбцы линейно независимы.

Систему столбцов $A_{j_1}, A_{j_2}, \dots, A_{j_r}$ называют *базисом* угловой точки $x^* = (x_1^*, x_2^*, \dots, x_n^*)$, переменные $x_{j_1}, x_{j_2}, \dots, x_{j_r}$ и координаты $x_{j_1}^*, x_{j_2}^*, \dots, x_{j_r}^*$ называются *базисными переменными* и *координатами* этой точки, остальные координаты $x_j^* = 0$ и соответствующие им переменные – *свободными*.

Если все базисные координаты угловой точки положительны, то точка называется *невырожденной*, если же среди базисных координат хотя бы одна равна нулю, то такая точка называется *вырожденной*. Из теоремы следует, что невырожденная угловая точка обладает единственным базисом, вырожденная угловая точка может обладать несколькими базисами. Следует иметь в виду, что в некоторых случаях наличие вырожденных угловых точек у многогранника решений может привести к заклиниванию симплекс-метода. В этом случае для решения задачи следует обратиться к дополнительной литературе.

4.3. Решение основной задачи линейного программирования

При решении основной задачи линейного программирования симплекс-методом вначале необходимо выполнить следующую подготовительную работу:

- ✓ найти ранг матрицы ограничений $r = \text{rang } A$;
- ✓ проанализировать систему ограничений: если $r < m$, то система содержит избыточные ограничения, которые должны быть выявлены и удалены; если $r = n$, то система ограничений допускает одно единственное решение задачи, которое можно найти методом Крамера или Гаусса; если $r = m > n$, то система ограничений допускает некоторое множество решений, оптимальное из которых может быть найдено симплекс-методом;
- ✓ найти начальный базис; другими словами, начальную угловую точку многогранника решений, содержащую r базисных и $n - r$ свободных переменных;
- ✓ построить каноническую форму для начального базиса, т.е. базисные переменные в системе ограничений выразить через свободные и исключить их из целевой функции.

Для определения начального базиса можно воспользоваться следующей процедурой.

1. Найти r линейно независимых столбцов матрицы ограничений;
2. Взять соответствующие переменные в качестве базисных;
3. Найти координаты соответствующей угловой точки, построив для нее каноническую форму;
4. Если все координаты неотрицательны, то искомая точка найдена, в противном случае полученный базис является недопустимым, находим следующую линейно независимую систему из r столбцов матрицы A и так далее.

Такой путь поиска начальной допустимой угловой точки в большинстве случаев достаточно трудоемкий и продолжительный. В этом случае можно воспользоваться следующей процедурой поиска начального допустимого базиса.

Вводятся m искусственных переменных $x_{n+1}, x_{n+2}, \dots, x_{n+m}$ и исходная задача приводится к виду

$$\begin{aligned} z &= c_1 x_1 + c_2 x_2 + \dots + c_n x_n \rightarrow \min; \\ a_{11} x_1 + a_{12} x_2 + \dots + a_{1n} x_n + x_{n+1} &= b_1; \\ a_{21} x_1 + a_{22} x_2 + \dots + a_{2n} x_n + x_{n+2} &= b_2; \\ &\dots \\ a_{m1} x_1 + a_{m2} x_2 + \dots + a_{mn} x_n + x_{n+m} &= b_m; \\ x_1, x_2, \dots, x_{n+m} &\geq 0. \end{aligned}$$

Искусственные переменные выбираются в качестве базиса, приведенная выше система является для него канонической формой, соответствующее решение $x_1 = x_2 = \dots = x_n = 0, x_{n+1} = b_1, x_{n+2} = b_2, \dots, x_{n+m} = b_m$ является допустимым. Далее с помощью симплекс-метода за m итераций все m искусственных переменных переводятся в свободные, в результате чего будет получен начальный допустимый базис для исходной задачи. После этого искусственные переменные из дальнейшего рассмотрения исключаются и с помощью симплекс-метода ищется оптимальное решение.

Иногда нет необходимости вводить все m искусственных переменных, достаточно их ввести только в некоторые ограничения.

После того, как найден начальный допустимый базис и для него построена каноническая форма, для поиска оптимального решения можно воспользоваться симплекс-методом.

4.4. Симплекс-метод

Пусть для начального допустимого базиса переменные x_1, x_2, \dots, x_r являются базисными, а $x_{r+1}, x_{r+2}, \dots, x_n$ - свободными. Тогда каноническая форма для него будет иметь следующий вид:

$$z = c_0 + c_{r+1}x_{r+1} + c_{r+2}x_{r+2} + \dots + c_n x_n;$$

$$x_1 = b_1 + a_{1,r+1}x_{r+1} + c_{1,r+2}x_{r+2} + \dots + c_{1n}x_n;$$

$$x_2 = b_2 + a_{2,r+1}x_{r+1} + c_{2,r+2}x_{r+2} + \dots + c_{2n}x_n;$$

...

$$x_r = b_r + a_{r,r+1}x_{r+1} + c_{r,r+2}x_{r+2} + \dots + c_{rn}x_n.$$

Первая таблица заполняется следующим способом.

№ итерации	Базис	Значение	$-x_{r+1}$...	$-x_r$...	$-x_n$
1	x_1	b_1	$a_{1,r+1}$...	a_{1r}	...	a_{1n}

	x_1	b_1	$a_{1,r+1}$...	a_{1r}	...	a_{1n}

	x_r	b_r	$a_{r,r+1}$...	a_{rr}	...	a_{rn}
	z	c_0	c_{r+1}	...	c_r	...	c_n

Все числа располагаются в верхнем левом углу. Коэффициенты a' и c' берутся с обратным знаком, на что указывает знак «-» перед переменными в заглавной строке. Клетку, где расположен коэффициент a'_{ij} , $i = \overline{1, r}$, $j = \overline{r+1, n}$, будем для краткости называть клеткой (ij) .

Далее работа с симплекс-таблицей осуществляется в следующем порядке:

1. Анализируется строка коэффициентов c'_i . Если все $c'_i \leq 0$, то задача решена и точка $x'_i = b'_i$ для базисных координат и $x'_i = 0$ для свободных является оптимальной, а c_0 - оптимальное значение функции. Если же существуют $c'_i > 0$, то переходим к шагу 2.

2. Определяется переменная для включения в базис: в строке x среди положительных коэффициентов c_i находим наибольший c_r ; столбец, содержащий c_s , выделяем; x_r - переменная для включения в базис.
3. Анализируется выделенный столбец (s). Если все $a_{is} \leq 0$, $i = \overline{1, r}$, то задача не имеет решения, иначе переходим на шаг 4.
4. Определяется переменная для исключения из базиса. С этой целью для всех положительных a_{is} вычисляются отношения b_i / a_{is} и выделяется строка (l), у которой это отношение наименьшее. x_l - новая свободная переменная.
5. Коэффициент a_{ls} , стоящий на пересечении выделенных строки и столбца, называется генеральным. Вычисляется коэффициент $v = 1 / a_{ls}$.
6. Заполняются нижние правые углы каждой клетки таблицы в следующем порядке:
 - 6.1. В клетке (ls) генерального коэффициента записывается число v .
 - 6.2. В клетках (lj), $j = \overline{r+1, n}$, и b_l выделенной строки записываются произведения верхних коэффициентов на v .
 - 6.3. В клетках js , $j = \overline{1, r}$, и клетке c_s выделенного столбца записываются произведения верхних элементов на $-v$.
 - 6.4. В остальных клетках ij , b_i , c_r , $i = \overline{1, r}$, $i \neq l$, $j = \overline{r+1, n}$, $j \neq s$, записывается произведение верхнего коэффициента клетки ij выделенной строки и нижнего клетки is выделенного столбца.
7. Заполняются верхние левые углы симплекс-таблицы следующей итерации:
 - 7.1. Новые базисная и свободная переменные меняются местами.
 - 7.2. Строка и столбец, соответствующие новым свободной и базисной переменным, заполняются нижними коэффициентами выделенных строки и столбца.
 - 7.3. В остальные клетки записываются суммы верхнего и нижнего коэффициентов, стоящих в соответствующих клетках предыдущей таблицы.

Литература

1. Тимковский В.Г. Дискретная математика в мире станков и деталей. – М.: Наука, 1992. – 145 с.
2. Майника Э. Алгоритмы оптимизации на сетях и графах. – М.: Наука, 1981. – 323 с.
3. Сакович В.А. Исследование операций. – Мн.: Выш. школа, 1985. – 256 с.
4. Костевич Л.С., Лапко А.А. Теория игр. Исследование операций. – Мн.: Выш. школа, 1982. – 230 с.
5. Барсов А.С. Линейное программирование в технико-экономических задачах. – М.: Наука, 1964.
6. Данциг Джордж Б. Линейное программирование, его применение и обобщения. – М.: Прогресс, 1966.
7. Банди Б. Основы линейного программирования. – М.: Радио и связь, 1989. – 176 с.
8. Методические указания к практическим работам по курсу «Основы математического моделирования» для студентов специальности 12.01 – «Технология машиностроения». – Мн.: БГПА, 1990.
9. Решетов Д.Н., Портман В.Т. Точность металлорежущих станков. – М.: Машиностроение, 1986. – 336 с.
10. Кучер А.М., Киватицкий М.М., Покровский А.А. Металлорежущие станки (альбом общих видов, кинематических схем и узлов). – Л.: Машиностроение, 1971. – 308 с.

Содержание

Введение.....	3
1. Метод математического моделирования.....	5
1.1. Сущность метода математического моделирования.....	5
1.2. Этапы решения задачи методом математического моделирования.....	6
1.3. Погрешности результатов при математическом моделировании.....	7
2. Лабораторные работы.....	8
Лабораторная работа №1. Оптимизация технологического маршрута механообработки.....	8
Лабораторная работа №2. Оптимизация оснащения магазина обрабатывающего центра.....	16
Лабораторная работа №3. Оптимизация последовательности переналадок технологической линии.....	21
Лабораторная работа №4. Оптимизация обработки на станке с ЧПУ.....	28
Лабораторная работа №5. Оптимизация последовательности горячей обработки.....	35
Лабораторная работа №6. Минимизация суммарного поля допуска.....	39
Лабораторная работа №7. Оптимизация раскроя промышленных материалов.....	44
Лабораторная работа №8. Математическое моделирование формообразования поверхностей деталей при обработке на металлорежущих станках.....	52
3. Некоторые положения дискретной математики.....	70
3.1. Некоторые понятия теории множеств.....	70
3.2. Основные понятия теории графов.....	70
3.3. Алгоритм Дейкстры поиска кратчайшего пути.....	71
3.4. Алгоритм поиска тончайшего пути.....	72
3.5. Алгоритмы поиска длиннейшего пути.....	73
3.6. Метод ветвей и границ.....	73
3.7. Алгоритм Литтла решения задачи о коммивояжере.....	75
3.8. Алгоритм ближайшего соседа.....	77

4. Основные положения теории линейного программирования.....	78
4.1. Решение общей задачи линейного программирования.....	78
4.2. Некоторые положения теории линейного программирования.....	79
4.3. Решение основной задачи линейного программирования.....	81
4.4. Симплекс-метод.....	83
Литература.....	85

Учебное издание

КЛИМОВИЧ Федор Федорович
ПРИСЕВОК Альберт Фомич

**МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ
ТЕХНОЛОГИЧЕСКИХ ЗАДАЧ В МАШИНОСТРОЕНИИ**

Учебно-методическое пособие по лабораторным работам
для студентов машиностроительных специальностей
высших учебных заведений

Редактор Т.Н.Микулик

Подписано в печать 29.02.2000.

Формат 60x84 1/16. Бумага тип. № 2. Офсет. печать.

Усл.печ.л. 5,1. Уч.-изд.л. 4,0. Тираж 100. Зак. 8.

Издатель и полиграфическое исполнение:

Белорусская государственная политехническая академия.

Лицензия ЛВ № 155 от 30.01.98. 220027, Минск, пр. Ф.Скорины, 65.