ДОПОЛНЕНИЕ К ВОПРОСУ 2

Классификация в зависимости от структуры и пространственно-временных свойств системы

1. Автоматические - реагируют на ограниченный набор внешних воздействий, внутренняя их организация приспособлена к переходу в равновесие при выводе их него.
2. Решающие – имеют постоянные критерии различия их постоянной реакции на широкие классы внешних воздействий.
3. Самоорганизующиеся – имеют гибкие критерии различения и гибкие реакции на внешние воздействия.
4. Предвидящие – характерно превосходящие сложности реакции системы, сложности внешних воздействий.
5. Превращающиеся – системы не связаны постоянной структурой, она изменяема.

Классификация системы по характеру связи между элементами

1. Детерминированные – наличие детерминированных связей
2. Стохастические – наличие стохастических связей

Классификация системы по характеру структуры управления

1. Централизованные – наличие управляющей системы
2. Децентрализованные – нет центра

ВОПРОС 3: Системный анализ в исследовании технических систем и процессов. Примеры

Основной предмет изучения Системного анализа – законы функционирования и свойства систем с управлением (системный анализ изучает только те системы, которыми можно управлять). Следовательно основные системы изучаемые Системным анализом - технические системы.

ВОПРОС 4: Принципы системного подхода

Системный подход – направление методологии научного познания и общественной практики, в основе которого лежит исследование объектов как систем.

Основные принципы системного подхода:

1. Целостность - позволяющая рассматривать одновременно систему как единое целое и в то же время как подсистему для вышестоящих уровней.
2. Иерархичность строения – то есть наличие множества элементов, расположенных на основе подчинения элементов низшего уровня элементами высшего уровня.
3. Структуризация – позволяет анализировать элементы системы и их взаимосвязи в рамках конкретной организационной структуры.
4. Множественность – позволяет использовать множество моделей для описания отдельных элементов и системы в целом.
5. Системность – свойство объекта обладать всеми признаками системы.

ВОПРОС 5: Определение понятия «модель». Классификация моделей. Общие требования к моделям. Структура моделей. Этапы моделирования

Модель – описание системы, отражающее определенную группу ее свойств. В процессе моделирования важно определить контекст модели, точку зрения модели и ее цель.

Контекст модели очерчивает границы моделирования системы, описывает ее взаимосвязи с внешней средой.

Цель модели отражает причину создания модели и определяет ее назначение.

Точка зрения модели – определяет позицию автора модели.

Классификация моделей:

* По области использования
  1. Учебные модели – наглядные пособия, тренажеры, обучающие модели
  2. Игровые модели – экономические, военные, деловые игры. Репетиция поведения объекта в различных ситуациях
  3. Исследовательские модели – создаются при исследовании процессов и явлений.
  4. Опытные модели – уменьшенные копии объектов
  5. Имитационные модели – отражающие реальность эксперименты.
* По фактору времени
  1. Статическая модель – одномоментный срез информации по объекту.
  2. Динамическая модель – позволяет увидеть изменения объекта во времени. Динамические модели тоже делятся:
     + Дискретные модели – изменяют свое состояние во времени скачкообразно
     + Непрерывные – изменяют свое состояние за сколь угодно малое приращение времени
     + Детерминированные модели – все воздействия и факторы известны на всем интервале управления
     + Стохастические модели – некоторые факторы носят случайный характер.
* По способу представления
  1. Материальные модели – воспринимают основные свойства оригинала и всегда имеют реальное воплощение.
  2. Информационная модель – совокупность информации, характеризующая свойства и состояние объекта, а также связь с внешним миром. Информационные модели делятся на:
     + Вербальную модель
     + Мысленная
     + Знаковая
     + Геометрическая
     + Математическая
     + Структурная
     + Логическая
     + Специальные

Основные требования к моделям:

1. Наглядность построения
2. Обозримость основных свойств и отношений
3. Доступность для исследования или воспроизведения
4. Простота исследования, воспроизведения
5. Сохранения информации содержащийся в оригинале и получение новой информации

Этапы моделирования:

1. Определение системы – установление границ, ограничений и измерителей эффективности системы подлежащей изучению
2. Формирование модели – переход от реальной системы к логической схеме
3. Подготовка данных необходимых для построения модели и представления их в соответствующей форме
4. Трансляция модели – описание модели
5. Оценка адекватности модели
6. Стратегическое планирование эксперимента
7. Тактическое планирование – определение способа проведения контроля испытаний предусмотренных планом эксперимента
8. Экспериментирование – процесс осуществления имитации с целью получения желаемых данных и анализа чувствительности
9. Интерпретация – построение выводов по данным полученным путем имитации
10. Реализация – практическое использование модели и результатов моделирования
11. Документирование – регистрация хода осуществления проекта и его результатов, а также документирования процесса создания и использования модели.

ВОПРОС 6: 6. Формализация задачи. Некоторые типовые проблемы, возникающие при исследовании. Интерполяция, экстраполяция, прогнозирование. Линейность и нелинейность. Дискретность и непрерывность. Детерминированность и случайность.

Формализация задачи - математическая постановка задачи, создание математической модели, завершающее этап постановки задачи. Формулируются система аксиом, описывающая не только сам объект, но некоторую алгебру, т.е. совокупность правил, определяющих допустимые операции над объектом. При формализации задачи должны быть определены функциональные зависимости, связывающие переменные и параметры модели.

Типовые проблемы, возникающие при исследовании:

Интерполяция – в вычислительной математике нахождение неизвестных промежуточных значений некоторой функции, по имеющемся дискретному набору ее известных значений определенным способом.

Экстраполяция – продолжение полученной в результате приближения функции за пределы интервала наблюдения.

Прогнозирование – специальное исследование, предметом которого является перспектива развития системы. Методы прогнозирования разделяются на статистические, причинно-следственные и комбинированные.

Линейность – это способность системы получить результаты прямо пропорциональные зафиксированным параметрам исследования.

Нелинейность – это особенность системы, процессы в которой описывают нелинейные дифференциальные уравнения.

Дискретность – состояние обратное непрерывности, при дискретности процесс изменяется между несколькими состояниями скачкообразно.

Непрерывность – свойство системы при котором малое приращение какого-либо аргумента в системе, приведет к малым приращением функций описывающих систему.

Детерминированность – процесс, исход которого полностью определен алгоритмом и значениями, заданными в начальном состоянии системы.

Случайность – процесс стохастического характера, когда некоторые параметры системы имеют случайный характер.

ВОПРОС 7: Графические способы функционального описания систем. Описание синтаксиса языка моделирования.

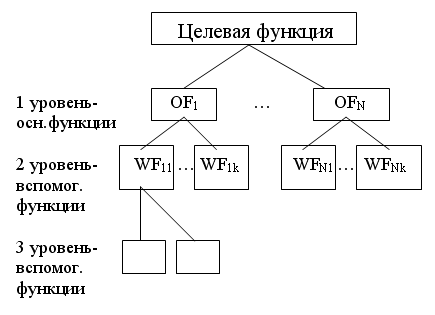
При анализе и синтезе систем используются графическое описание, разновидностями которого являются:

* Дерево функций системы
* Стандарт функционального моделирования IDEF0

Все функции, реализуемые сложной системой, могут быть условно разделены на три группы:

* Целевая функция – отражает сущность и смысл существования системы
* Основные функции – обеспечивают условия выполнения целевой функции
* Дополнительные функции – обеспечивают условия выполнения основных функций

Дерево функций:



Описание методологии IDEF0:

Описание IDEF0 модели построено в виде иерархической пирамиды, в вершине которой представляется самое общее описание системы, а основание представляет собой множество более детальных описаний.

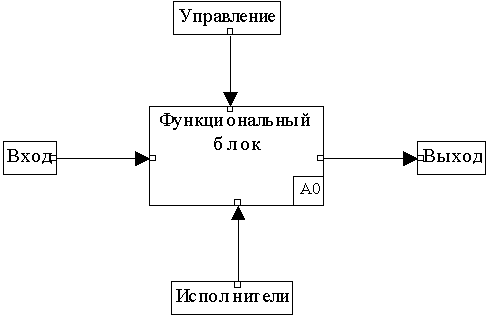
IDEF0 методология построена на следующих принципах:

* Графическое описание моделируемых процессов. Графический язык Блоков и Дуг IDEF0 Диаграмм отображает операции или функции в виде блоков, а взаимодействие между выходами операций, входящими в Блок или выходящими из него Дугами
* Лаконичность. За счет использования графического языка описания процессов достигается одной стороны точность описания, а с другой – краткость.

Блоки служат для отображения функций (действий) выполняемых моделируемой системой. Сформулированные функции должны содержать глагольный оборот:

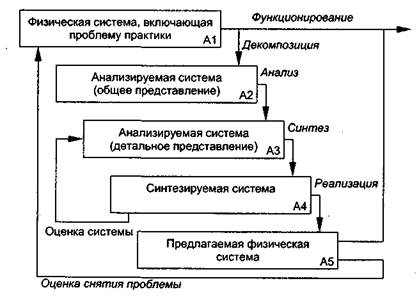
* *Глагол + объект действия + (дополнение)*

Дуги служат для отображения информации или материальных объектов, которые необходимы для выполнения функции или появляются в результате ее выполнения.

Место соединения Дуги с Блоком определяет тип интерфейса. 

ВОПРОС 8: Общий подход к решению проблемы. Основные задачи системного анализа. Стратегии декомпозиции. Этап синтеза системы. Формирование общего представления системы. Формирование детального представления системы

Общий подход к решению проблем может быть представлен как цикл:



Основные задачи системного анализа могут быть представлены в виде трёхуровневой дерева функций:



Рассмотрим некоторые наиболее часто применяемые стратегии декомпозиции:

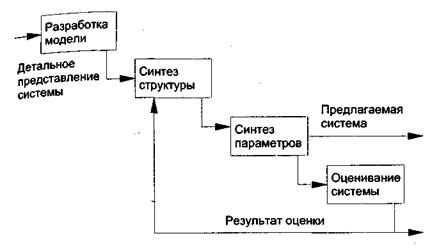
Функциональная декомпозиция – базируется на анализе функций системы. При этом ставится вопрос что делает система, независимо от того, как она работает. Основанием разбиения на функциональные подсистемы служит общность функций, выполняемых группами элементов.

Декомпозиция по жизненному циклу. Признак выделения подсистем – изменения закона функционирования подсистем на разных этапах цикла существования системы «от рождения до гибели». Рекомендуется применять эту стратегию, когда целью системы является оптимизация процессов и когда можно определить последовательность стадии преобразования входов в выходы.

Декомпозиция по физическому процессу. Признак выделения подсистем – шаги выполнения алгоритма функционирования подсистемы, стадии смены состояний. Хотя эта стратегия полезна при описании существующих процессов, результатом ее часто может стать слишком последовательное описание системы, которое не будет в полной мере учитывать ограничения, диктуемые функциями друг другу. При этом может оказаться скрытой последовательность управления. Применять эту стратегию следует, только если целью модели является описание физического процесса как такового.

На этапе синтеза осуществляются:

1. Разработка модели требуемой системы (выбор математического аппарата, моделирование, оценка модели по критериям адекватности, простоты, соответствия между точностью и сложностью, баланса погрешностей, многовариантности реализаций, блочности построения)
2. Синтез альтернативных структур системы, снимающей проблему.
3. Синтез параметров системы, снимающей проблему
4. Оценивание вариантов синтезированной системы ( обоснование схемы оценивания, реализация модели, проведение эксперимента по оценке, обработка результатов оценивания, анализ результатов, выбор наилучшего варианта)



Формирование общего представления системы.

Стадия 1: Выявление главных функций (свойств, целей, предназначения) системы. Формирование (выбор) основных предметных понятий, используемых в системе.

Стадия 2: Выявление основных функций и частей (модулей) в системе. Понимание единства этих частей в рамках системы.

Стадия 3: Выявление основных процессов в системе, их роли, условий осуществления; выявление стадийности, скачков, смен состояний в функционировании; в системах с управлением – выделение основных управляющих факторов.

Стадия 4: Выявление основных элементов «несистемы», с которыми связана изучаемая система. Выявление характера этих связей.

Стадия 5: Выявление неопределенностей и случайностей в ситуациях их определяющего влияния на систему (для стохастических систем).

Стадия 6: Выявление разветвленной структуры, иерархии, формирование представлений о системе как о совокупности модулей, связанных входами-выходами.

Формирование детального представления системы.

Стадия 7: Выявление всех элементов и связей, важных для целей рассмотрения. Их отнесение к структуре иерархии в системе. Ранжирование элементов и связей по их значимости.

Стадия 8: Учет изменений и неопределенностей в системе.

Стадия 9: Исследование функций и процессов в системе в целях управления ими. Введение управления и процедур принятия решения

Вопрос 9: Виды моделирования. Принципы и подходы к построению математических моделей. Этапы построения математической модели.

Виды моделирования:

Информационное моделирование – совокупность информации, характеризующая существенные свойства и состояния объекта, процесса, явления, а также взаимосвязь с внешним миром.

Компьютерное моделирование - процесс вычисления компьютерной (численной) модели на одном или нескольких вычислительных узлах. Реализует представление объекта в форме отличной от реальной, но приближенной к алгоритмическому описанию.

Математическое моделирование – математическое представление объекта, исследование которого позволяет получать информацию о некотором другом объекте.

Логическое моделирование – модель в представлении знаний, вся информация необходимая для решения прикладных задач, рассматривается как совокупность фактов и утверждений, которые представляются как формулы в некоторой логике.

Физические моделирование – метод экспериментального изучения различных физических объектов или явлений, основанный на использовании модели, имеющей ту же физическую природу, что и изучаемый объект.

Принципы и подходы к построению математических моделей:

1. Адекватность – соответствие модели целям исследования по уровню сложности и организации, а также соответствие реальной системе относительно выбранного множества свойств.
2. Соответствие модели реальной задаче. Модель должна строиться для определенного класса задач или конкретной задачи исследования системы.
3. Упрощение при сохранении существенных свойств системы. Модель должна быть в некотором отношении проще прототипа.
4. Соответствие между требуемой точностью результатов моделирования и сложностью модели. Модели всегда носят приближенный характер.
5. Баланс погрешности различных видов.
6. Многовариантность реализации элементов модели. Разнообразие реализации одного и того же элемента отличающихся по точности, обеспечивает соотношение «точность/сложность»
7. Блочное строение. Облегчение разработки сложных моделей и появление возможности использовать накопленный опыт и готовые блоки с минимальными связями между ними.

Этапы построения математической модели

1. Содержательное описание моделируемого объекта.
2. Формализация операций.
3. Проверка адекватности модели.
4. Корректировка модели.
5. Оптимизация модели.

Вопрос 10: Выбор решения при системном анализе и моделировании технических систем. Формализация задачи принятия решения.

Решение – действие над множеством альтернатив, в результате которого получается подмножество выбранных альтернатив. Обычно это один вариант, одна альтернатива, но не обязательно.

Задача принятия решения состоит в формировании множества возможных вариантов, обеспечивающих разрешение проблемной ситуации при существующих ограничениях, и выделении среди этих вариантов одного лучшего или нескольких предпочтительных вариантов, удовлетворяющих предъявляемым к ним требованиям. Формально задачу принятия решения D можно записать в следующем обобщенном виде:



Где

F — формулировка задачи принятия решения, которая включает в себя содержательное описание стоящей проблемы и при необходимости ее модельное представление, определение цели или целей, которые должны быть достигнуты, а также требования к виду окончательного результата;

А — совокупность возможных вариантов (альтернатив), из которых производится выбор. Это могут быть реально существующие варианты, в качестве которых в зависимости от контекста задачи выступают объекты, кандидаты, способы достижения цели, действия, решения и т. п., либо гипотетическое множество всех теоретически возможных вариантов, которое может быть даже бесконечным. Выбор возникает только тогда, когда имеется не менее двух возможных вариантов решения проблемы;

X — совокупность признаков (атрибутов, параметров), описывающих варианты и их отличительные особенности. В качестве признаков выступают, во‑первых, объективные показатели, которые характеризуют те или иные свойства, присущие вариантам, и которые, как правило, можно измерить; во‑вторых, субъективные оценки, которые обычно даются по специально отобранным или сконструированным критериям, отражающим важные для участников выбора черты вариантов.

G — совокупность условий, ограничивающих область допустимых вариантов решения задачи. Ограничения могут быть описаны как содержательным образом, так и заданы в виде некоторых формальных требований к вариантам и/или их признакам. Например, это могут быть ограничения на значения какого- либо признака или различная степень характерности (выраженности) признака для тех или иных вариантов, или невозможность одновременного сочетания определенных значений признаков для реально существующих вариантов.

Р — предпочтения одного или нескольких ЛПР, которые служат основой для оценки и сравнения возможных вариантов решения проблемы, отбора допустимых вариантов и поиска наилучшего или приемлемого варианта. Достаточно часто для упрощения постановки задачи принятия решения часть информации, описывающей предпочтения ЛПР, превращается в ограничения.

Вопрос 11: Постановка задачи принятия решений. Декомпозиция задачи принятия решения и оценка альтернатив.

Выделение свойств альтернатив является задачей декомпозиции. Декомпозиция в общем случае имеет иерархический характер. Каждое свойство 1-го уровня делится на набор свойств 2-го уровня и так далее до такого уровня, на котором свойства оказываются легко сравнимыми.

Используются три способа сравнения альтернатив по их свойствам:

* попарное (групповое) сравнение по определенному свойству;

Формально — это бинарная операция по признаку R.

xiRxj — означает, что согласно признаку R альтернатива xi предпочтительней альтернативы xj.

* на основе естественных или искусственно введенных числовых характеристик свойств;

Свойства, для которых известны числовые характеристики, называются критериями. При идеальной декомпозиции имеет место набор критериев, т.е. имеет место многокритериальная задача.

ВОПРОС 12: Понятие абстрактной системы. Задачи анализа и синтеза. Структурные схемы

Абстрактные системы - это средства для характеристики реальных процессов и явлений или все множество логических моделей.

Свойства:

* Целостность – любая система состоит из функционального объединения связей, подсистем и элементов. Объединение элементов в единое целое придает системе качество. Принципиально отличается от простой суммы свойств, составляющих элемент.
* Динамизм – наличие материальных потоков переноса в рамках системы вещества, энергии, информации.
* Многофункциональность – выполнение системой своего предназначения требует решения многих частных задач.
* Иерархичность – организация сложной системы сопряжена с обязательным введением принципа подчиненности.
* Информационность – за расчет информационных потоков, циркулируемых в системе, достигается согласованность функционирования всех подсистем.
* Управляемость – свойство является базовым в организации системы. Для достижения цели системой необходимо управление.

Под анализом понимается процесс исследования системы управления, ос­нованный на ее декомпозиции с последующим определением статических и динамических характеристик составляющих элементов, рассматриваемых во взаимосвязи с другими элементами системы и окружающей средой

Цели анализа системы управления:

* детальное изучение системы управления для более эффективного использования и принятия решения по ее дальнейшему совершенствованию или замене;
* исследование альтернативных вариантов вновь создаваемой системы управления с целью выбора наилучшего варианта.

К задачам анализа системы управления относятся:

* определение объекта анализа;
* структурирование системы;
* определение функциональных особенностей системы управления;
* исследование информационных характеристик системы;
* определение количественных и качественных показателей системы управления;
* оценка эффективности системы управления;
* обобщение и оформление результатов анализа.

Под синтезом понимается процесс создания новой системы путем определения ее рациональных или оптимальных свойств и соответствующих показателей.

Цели синтеза системы управления: создание новой системы управления на основе новых достижений науки и техники; совершенствование существующей системы управления на основе выявленных недостатков, а также появлений новых задач и требований.

Задачи синтеза систем управления заключаются в определении структуры и параметров системы исходя из заданных требований к значениям показателей эффективности ее функционирования, а также способов обеспечения целей функционирования системы.

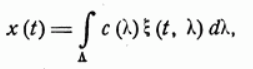
Задачи синтеза:

* формирование замысла и цели создания системы управления;
* формирование вариантов новой системы;
* приведение описаний вариантов системы во взаимное соответствие;
* оценка эффективности вариантов и принятия решения о выборе варианта новой системы;
* разработка требований к системе управления;
* разработка программ реализации требований к системе управления;
* реализация разработанных требований к системе управления.

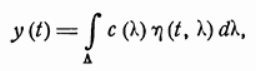
Структурная схема — это совокупность элементарных звеньев объекта и связей между ними, один из видов графической модели. Под элементарным звеном подразумевается часть объекта, системы управления и т. д., которая реализует элементарную функцию.

ВОПРОС 13: Весовые Функции линейных систем и их преобразования при соединениях систем (параллельном, последовательном и с обратной связью)

Если входное возмущение системы выражается интегралом:

 (1)

То выходная переменная системы y(t) выражается точно таким же интегралом

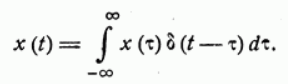
 (2)

Где  представляет собой выходные переменные данной системы, соответствующие возмущению  при тех же значениях параметра  :

 (3)

Где A – оператор системы

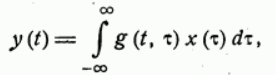
Рассмотрим систему с одним входом и одним выходом. Пользуясь  , можно любую функцию x(t) выразить интегралом:

(4)

Эта формула является частным случаем формулы для x(t), где

(5)

Таким образом формула (4) дает разложение любой функции x(t) на элементарные импульсы. На основе формул (2), (3) и (5) выходная переменная линейно системы выражается формулой:

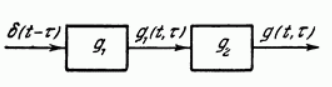
(6)

Где  - реакция этой системы в тот момент t га возмущение , т.е. на единичный импульс, действующий в момент :

(7)

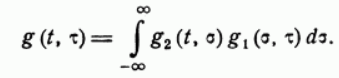
Функция  называется весовой функцией или импульсной переходной функцией линейно системы. Зная весовую функцию линейной системы, имеющей один вход и один выход можно по формуле определить выходную переменную этой системы при произвольном возмущении x(t). Таким образом, весовая функция линейной системы полностью характеризует ее динамические свойства.

Рассмотрим последовательность соединения двух линейных систем имеющих весовые функции g1 и g2.



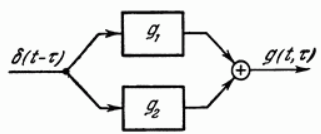
(Получится линейная система)

Для определения весовой функции этой системы приложим к ее входу единичный импульс . Тогда получим на выходе первой системы величину  , а на выходе второй системы – искомую весовую функцию последовательного соединения. Применяя для вычисления выходной переменной формулу (6) получим следующее:

 (8)

Если бы g1 и g2 были поменяны местами в соединении, то они поменяют место и в формуле, при этом сама g тоже бы поменялась.

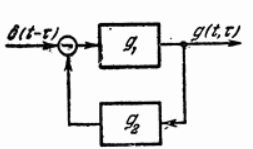
Рассмотрим теперь параллельное соединение линейных систем.



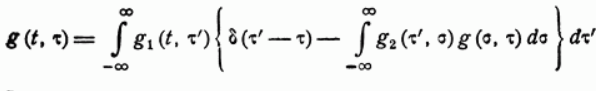
(Все равно линейная система)

Весовая функция параллельного соединения равна сумме весовых функций соединяемых в системе:  
(9)

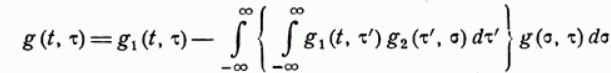
Рассмотрим задачу определения весовой функции линейной системы с обратной связью.



При подаче на вход единичного импульса  ее выходная переменная, совпадающая с весовой функцией g1, будет равна искомой функции g. Эта же величина поступит на вход системы с весовой функцией g2, находящейся в цепи обратной связи. Ее выходная переменная будет вычитаться из входного сигнала , и получается разность будет входным возмущением для системы с весовой функцией g1. Получаем следующее:

(10)

Выполняя интегрирование и изменяя порядок интегрирования в двойном интеграле, приведем к виду:

(11)

Получается, для определение весовой функции линейной системы с обратной связью, необходимо решить интегральное уравнение (11).

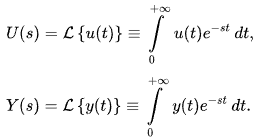
Вопрос 14: Передаточные Функции линейных стационарных систем. Нахождение передаточных Функций для систем, заданных дифференциальными уравнениями и структурными схемами.

Пусть u(t) - входной сигнал Линейной стационарной системы, y(t) - ее выходной сигнал. Тогда передаточная функция W(s) этой системы записывается в виде:



Где  - оператор передаточной функции преобразования Лапласа.

Y(s) и U(s) – преобразования Лапласа для сигналов u(t) и y(t) соответственно:

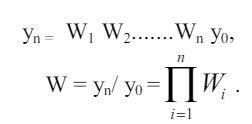


Правила преобразования для структурных схем.

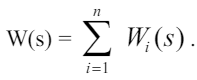
1. Последовательное соединение звеньев. При последовательном соединении выходная величина каждого предшествующего соединения является входом следующего. Группу последовательно включенных звеньев можно заменить одним сложным звеном, передаточная функция которого будет равна произведению передаточных функций отдельных звеньев:



От сюда следует:



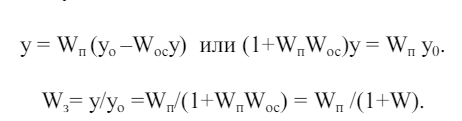
1. Параллельное соединение звеньев. При параллельном соединении, на вход каждому звену подается один и тот же сигнал, а выход каждого суммируется. Следовательно, имеет место:

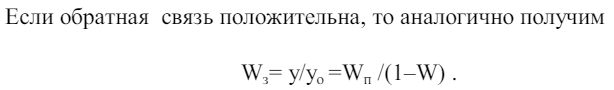


1. Звено с обратной связью. Считается, что звено охвачено обратной связью, если его выходной сигнал через какое-либо звено подается на вход. При этом, если сигнал обратной связи y1 вычитается из входного сигнала y0, то обратная связь называется отрицательной. Если сигнал обратной связи y1 складывается с входным воздействием y0, то обратная связь называется положительной.

Передаточная функция такой разомкнутой цепи W равна произведению передаточной функции Wп прямой цепи и передаточной функции Wо.с. обратной связи: W=WпWо.с Передаточная функция W замкнутой цепи с отрицательной (положительной) обратной связью равна передаточной функции прямой цепи, деленной на единицу плюс (минус) передаточная функция разомкнутой цепи. Получаем:

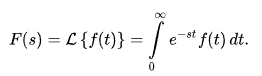




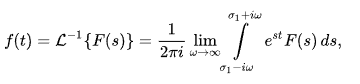


ВОПРОС 15: Применение преобразования Лапласа для нахождения реакции систем на заданные воздействия. Реакция на ступенчатое воздействие и показатели качества.

Прямое преобразование Лапласа



Обратное преобразование Лапласа:



Пример: Определить реакцию системы с передаточной функцией ****** на произвольное входное воздействие **.

Решение: Определим реакцию системы с использованием преобразований Лапласа.

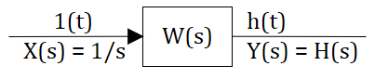
Выходной сигнал имеет вид ******.

Изображение входного сигнала найдем с помощью таблицы: .

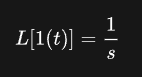
Изображение выходного сигнала имеет вид: ******.

Применим обратное преобразование Лапласа к изображению выходного сигнала и получим оригинал сигнала во временной области: **

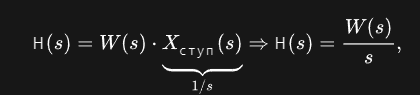
Реакция системы на ступенчатое воздействие



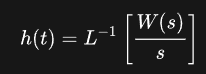
Для единичного ступенчатого воздействия преобразование Лапласа тоже известно



тогда в изображениях получаем, что реакция системы Н(s) на ступенчатое воздействие, рассчитывается так:



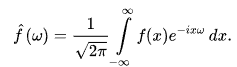
Реакция системы на единичное ступенчатое воздействие рассчитывается обратным преобразованием Лапласа:



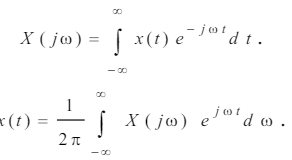
ВОПРОС 16: Применение преобразования Фурье для нахождения реакции систем на гармонические воздействия. Частотные показатели качества.

Преобразование Фурье – операция, сопоставляющая одной функции вещественной переменной другую функцию, которая описывает коэффициенты(«амплитуды») при разложении исходной функции на элементарные составляющее – гармонические колебания с разными частотами.

Преобразование Фурье функции f вещественной переменной является интегральным и задается следующей формулой.



Пусть на вход линейной стационарной системы с частотной передаточной функцией подан произвольный сигнал , обладающий спектральной характеристикой . Как известно, функции  связаны формулами прямого и обратного преобразования Фурье.

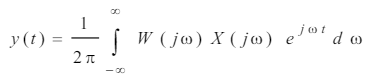


Согласно спектральному методу анализа спектральная характеристика  сигнала  на выходе системы равна:



Применив к полученному выражению обратное преобразование Фурье, найдем

выходной сигнал системы



Таким образом, сигнал на выходе линейно стационарной системы можно получить суммированием спектра  входного сигнала, взятых с весом . Другими словами, частотная передаточная функция  играет роль весовой функции, определяющей относительный вклад различных составляющих спектра  в выходной сигнал .

Рассмотрим частотные критерии качества, позволяющие судить о динамических свойствах системы по ее частотным характеристикам. К их числу могут быть отнесены запасы устойчивости САУ по усилению и фазе, которые могут быть определены по АФХ или логарифмическим амплитудно- и фазо-частотной характеристикам системы в разомкнутом состоянии.

Для оценки качества переходного процесса минимально-фазовой системы достаточно знать вид ее амплитудно-частотной характеристики . С целью обеспечения сопоставимости значений критериев для различных САУ характеристика нормируется, для чего ее ординаты делятся на начальное значение , т.е. на ее значение при :

.

При этом АЧХ нормированных статических систем  начинается с единицы (рис. 4.2).

К частотным показателем качества, определяемым по, относятся:

* полоса пропускания системы;
* резонансная частота :
* показатель колебательности *M*.

Вопрос 17: Частотные передаточные функции и частотные характеристики линейных стационарных систем. Критерий устойчивости Найквиста и понятие запаса устойчивости.

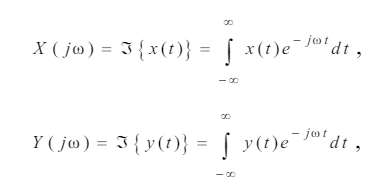
Рассмотрим линейную стационарную систему, описываемую дифференциальным

Уравнением



Применим к дифференциальному уравнению системы преобразование Фурье.

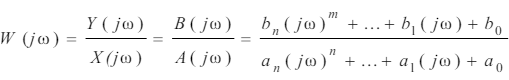
Имея в виду, что



и используя свойство линейности и формулы для изображений производных, получим



Частотная передаточная функция W( jω )определяется как отношение преобразований Фурье выходного и входного сигналов:



Функция W( jω )характеризует динамические свойства самой системы и не зависит от характера приложенных к системе воздействий. Она легко может быть получена

из передаточной функции при чисто мнимых значениях переменной .

Для наглядного представления частотных свойств системы используются так называемые частотные характеристики.

Вектор-годограф W( jω ), построенный на комплексной плоскости при изменении

частоты от ω = 0 до ω = inf, называется амплитудно-фазовой частотной характеристикой системы.

Представим функцию W( jω ) в следующем виде



Где



Функция  называется амплитудно-частотной характеристикой системы и представляет собой отношение амплитуды установившегося гармонического сигнала к амплитуде выходного гармонического сигнала при частоте входного сигнала, равной ω.

Функция , называется фазо-частотной характеристикой системы. Она показывает, на сколько выходной сигнал y(t) при данной частоте ω сдвинут по фазе (углу) относительно входного сигнала x(t).

Критерий устойчивости Найквиста - замкнутая САУ устойчива если при изменении ω от 0 до inf А.Ф.Х разомкнутой системы годограф W(j ω) охватывает k/2 раз точку (-1, j0) в положительном направлении, где k – число корней характеристического уравнения разомкнутой системы, лежащий в правой полуплоскости.

При работе параметры системы по разным причинам могут меняться в некоторых пределах. Эти колебания параметров приводят к потере устойчивости системы, если она работает вблизи границы устойчивости. Обычно проектируют систему так, чтобы она работала вдали от границы устойчивости. Степень этого удаления называют запасом устойчивости.

Согласно критерия Найквиста, чем дальше АФЧХ от критической точки (-1; j0),тем больше запас устойчивости. Различают запасы устойчивости по модулю (амплитуде) и по фазе.

ВОПРОС 18: Понятие динамической системы по Калману. Уравнения линейных систем в пространстве состояний.

Динамическая система — множество элементов, для которого задана функциональная зависимость между временем и положением в фазовом пространстве каждого элемента системы. Данная математическая абстракция позволяет изучать и описывать эволюцию систем во времени.

Состояние динамической системы в любой момент времени описывается множеством вещественных чисел (или векторов), соответствующим определённой точке в пространстве состояний. Эволюция динамической системы определяется детерминированной функцией, то есть через заданный интервал времени система примет конкретное состояние, зависящее от текущего.

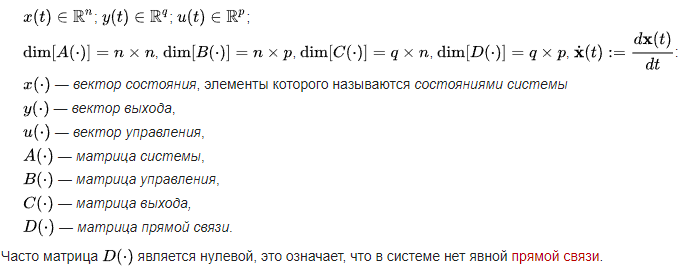
Пространство состояний обычно называют фазовым пространством динамической системы, а траекторию движения изображающей точки в этом пространстве — фазовой траекторией.

Линейные непрерывные системы:

Для случая линейной системы с 𝑝 входами, 𝑞 выходами и 𝑛 переменными состояния описание имеет вид:

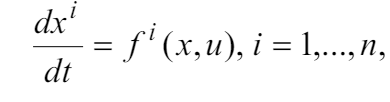


Где

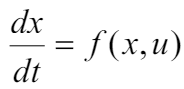


ВОПРОС 24: Постановка задачи оптимального управления.

Закон движения фазовой точки и закон воздействия управления записывается в виде системы дифференциальных уравнений:

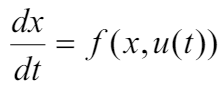


Или в векторной форме:

 (1)

где функции  непрерывны по переменным x и u, непрерывно дифференцируемы по переменной x. Здесь рассматривается случай, когда система автономна, то есть правые ее части не зависят явно от времени t.

Рассмотрим произвольное допустимое управление u(t). Перепишем уравнение в следующем виде:

 (2)

Тогда при любых начальных условиях  однозначно определяется траектория движения объекта , то есть решение этого уравнения, определенное на некотором отрезке времени

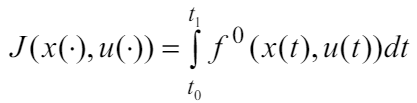
Назовем его решением системы (2), соответствующим управлению u(t) при начальном условии 

Будем говорить, что допустимое управление u(t),  переводит фазовую точку x из положения , если решение x(t) уравнения (2) с начальным условием 

определено на .

Такую пару (x(t),u(t)) назовем управляемым процессом, определенном на отрезке 

Пусть задана еще одна функция  непрерывная по переменным x и u, непрерывно дифференцируемая по переменной x. Приведем формальную постановку задачи оптимального управления.

Найти среди всех допустимых управлений, переводящих фазовую точку из положения  в положение , такое, для которого функционал 

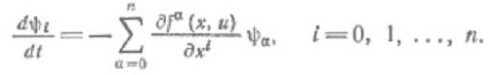
принимает наименьшее значение.

Вопрос 25: Принцип максимума Л.С. Понтрягина.

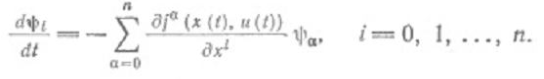
Для формулировки теоремы рассмотрим систему уравнений:

(1)

Рассмотрим еще систему уравнений относительно вспомогательных переменных  :

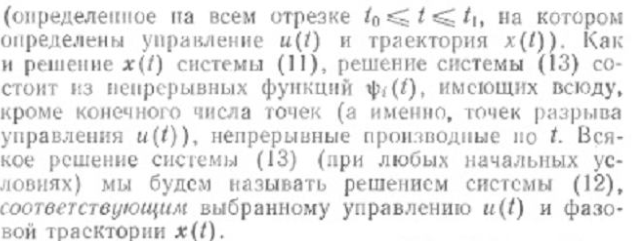
(2)

Если мы выбрали некоторое допустимое управление u(t), , и имеем соответствующую фазовую траекторию x(t) системы (1) с начальным условием  то система (2) принимает вид



Эта система линейна и однородна, поэтому при любых начальных условиях для , она допускает единственное решение

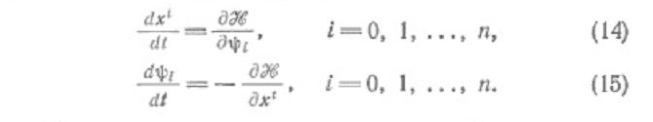


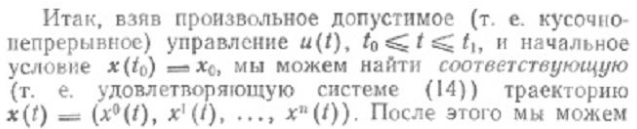
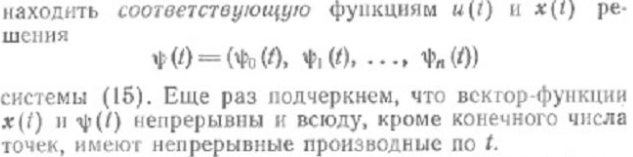


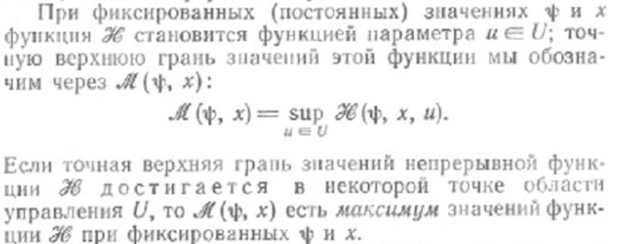
Объединим системы (1), (2) одной записью, для чего рассмотрим следующую функцию переменных 

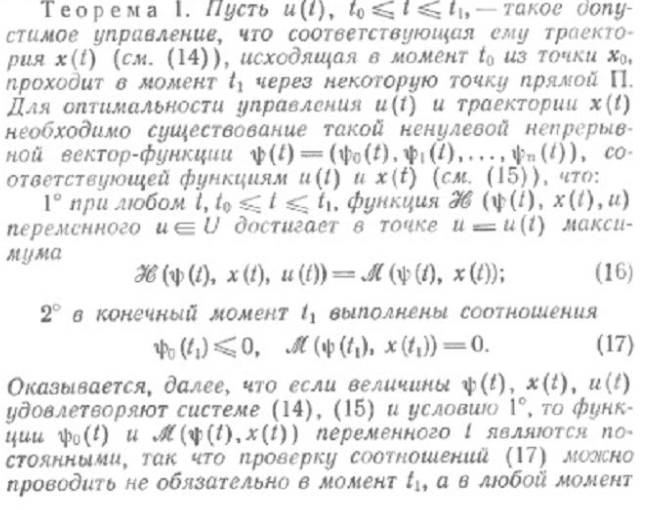


Проверяется что написанные выше системы (1) и (2) могут быть с помощь этой функции записаны в виде гамильтоновой системы:







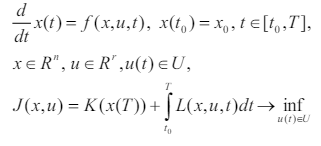


Вопрос 26: Метод динамического программирования. Численное решение уравнений динамического программирования.

Динамическое программирование

В основу динамического программирования положен принцип оптимальности. Согласно этому принципу оптимальное управление определяется конечной целью управления и состоянием системы в рассматриваемый момент времени, независимо от того, каким образом система пришла в это состояние, т.е. оптимальное управление не зависит от предыстории системы. Это означает, что для любой оптимальной траектории каждый участок, связывающий любую промежуточную точку этой траектории с конечной, также является оптимальной траекторией.

Пусть задача управления выглядит следующим образом. Задан динамический объект и функционал качества



(4.2)

(4.1)





Численное решение уравнений динамического программирования

Для решения уравнение



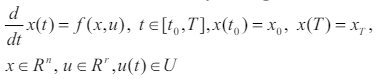
(4.8)

С критерием



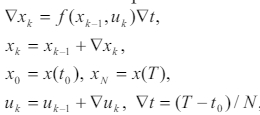
(4.9)

Применяют численное решение в виде многошагового процесса. Исходное уравнение вида:



(4.13)

Заменяют конечо-разностным уравнением:



(4.14)

где k=1,2,…,N; N – число принятых расчетных интервалов (шагов).

Функционал качества для данной задачи записывается в виде



(4.15)

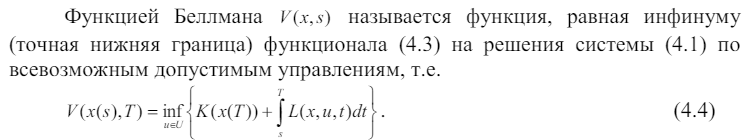
При численном решении задачи оптимизации участки процесса рассматриваются в последовательности, обратной их номеру, т.е. от конца процесса к началу.

Рассмотрим начало последнего участка процесса, т.е. момент времени. Этому моменту соответствует значение координаты . Согласно принципу оптимальности необходимо, чтобы было выбрано оптимальное управление на последнем интервале , переводящее объект из состояния  в . Выбор значений  влияет лишь на последний член суммы:



(4.16)

Выберем такое  при котором будет обеспечен минимум полученной сумме. При этом на основании (4.4)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Получаем



В результате на первом шаге назад (последнем интервале) определим значения . Далее переходим к второму шагу назад.

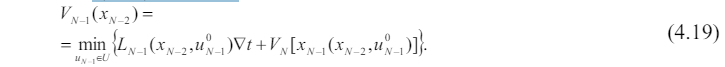
Важно:



Так как согласно (4.15)



То вместо (4.18) запишем



Общая формула для этого:



Таким образом, все расчеты проводятся «шаг за шагом назад». В результате получим значения , по которым можно построить приближенную оптимальную траекторию  и найти приближенное оптимальное управление . Таким образом, в результате решения задачи численным методом получим квазиоптимальное управление.

Вопрос 27: Оптимальное управление линейными объектами. Задача со свободным правым концом и заданным временем окончания переходного процесса.

Рассматриваться управляемые динамические системы, описываемые линейными дифференциальными уравнениями:

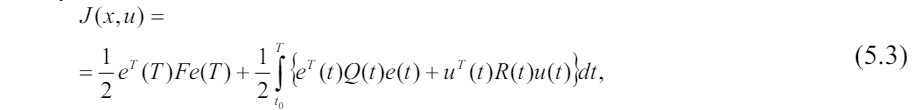


Где  - состояние системы; - управляющий вход системы;  - выход системы. Матрицы  размерностями  соответственно. Требуется отыскать такое управление u(t), при котором ошибка системы



Была бы «малой».

Будем рассматривать критерий следующего типа:



Где F, Q(t) – положительно полуопределенные матрицы размерностью m x m;

R(t) – положительно определенная матрица размерностью r x r.

Рассмотрим каждый член функционала (5.3). Так как Q(t) положительно полуопределенная, то этот член неотрицателен при любом e(t). Так как R(t)- положительно определенная матрица, то этот член положителен при любых u(t)!=0.

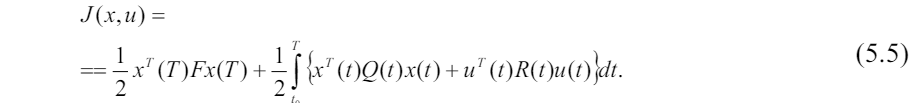
Критерий качества (5.3) удобен математически, и его минимизация приводит к тому, что оптимальные системы оказываются линейными.

Задача со свободным правым концом и заданным временем окончания переходного процесса

Рассмотрим линейную управляемую динамическую систему с переменными параметрами



И функционал



Требуется найти достаточно экономичное управление, которое удерживало бы систему вблизи нуля. Будем считать, что управление u(t) не ограничено, время Т задано, матрицы Q(t) и F положительно полуопределены, матрица R(t) положительно определена.

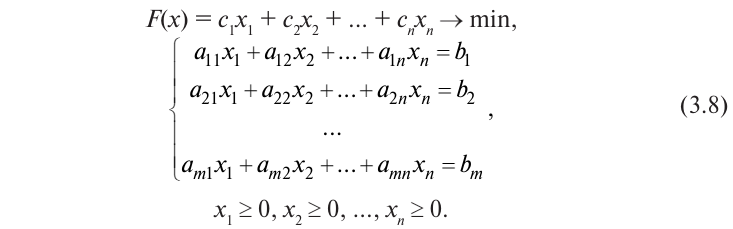
Глава 5 АФАНАСЬЕВ

Вопрос 28: Дифференциальные игры. Постановка задачи. Линейные игры преследования с квадратичным функционалом.

Глава 7 АФАНАСЬЕВ

Вопрос 29: Постановка и основные свойства задач линейного программирования

Задача минимизации линейной целевой функции F(x) на множестве ограничений, заданных системой линейных равенств при условии неотрицательности переменных, является задачей линейного программирования (ЗЛП) в каноническом виде. Формально ЗЛП в каноническом виде записывается так:



Всякое неотрицательное решение  системы (3.1) называется допустимым решением задачи. Неотрицательное (допустимое) решение  системы (3.1), минимизирующее целевую функцию F(x), называется оптимальным решением.

Правила приведения ЗЛП к каноническому виду:

1. Переход от задачи минимизации целевой функции к задаче ее максимизации. Для этого достаточно ввести новую целевую функцию . Очевидно, что переменные, минимизирующие  максимизируют F(x)
2. Преобразование неравенства в равенство путем добавления новой переменной.

Допустим, ограничение имеет вид: 

Перенесем все элементы в правую сторону и введем новую (добавочную) переменную  положив



Условие  накладываемое на любое значение переменной, входящее в область допустимых значений, равносильно неравенству. Итак, неравенство (3.2) эквивалентно равенству (3.4) при условии .

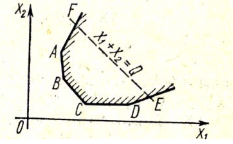
1. Если какая-либо переменная , то в рассмотрение вводится новая переменная . Если переменная может менять знак, то исходная задача разбивается на две: в одной задаче , в другой наоборот. Каждая задача решается по отдельности, и из двух решений выбирается лучшее (минимальное или максимальное).

СВОЙСТВА РЕШЕНИЙ ЗАДАЧ ЛИНЕЙНОГО ПРОГРАММИРОВАНИЯ

Теорема 1: Множество всех планов задачи линейного программирования выпукло.

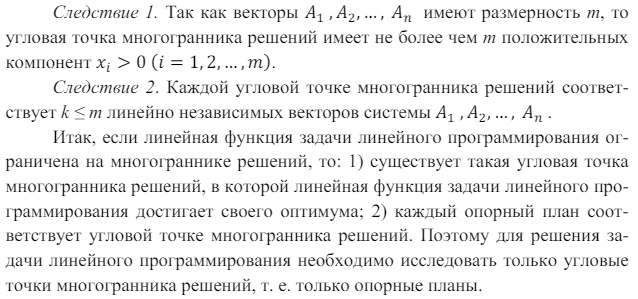
Теорема 2: Линейная функция задачи линейного программирования достигает своего минимального значения в угловой точке многогранника решений. Если линейная функция принимает минимальное значение более чем в одной угловой точке, то она достигает того же значения в любой точке, являющейся выпуклой линейной комбинацией этих точек.

Замечание. Если многогранник решений – неограниченная область, то вводят в систему дополнительное ограничение  , где Q – достаточно большое число. Введение этого ограничения равносильно отсечению гиперплоскостью  от многогранной неограниченной области ограниченного многогранника, для точек которого теорема 2 уже выполняется.



Теорема 3. Если известно, что система векторов линейно независима и такова, что  где все  , то точка является угловой точкой многогранника решений.

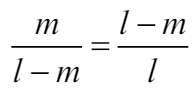
Теорема 4. Если – угловая точка многогранника решений, то векторы в разложениисоответствующие положительным  являются линейно независимыми.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

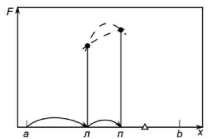
Вопрос 30: Методы золотого сечения и Фибоначчи

Методы золотого - Этот алгоритм используется для нахождения минимума функции.

Отрезок длины l делится на две части m и l – m так, чтобы меньшая часть относилась к большей, как большая ко всему отрезку:



1. Рассмотрим отрезок [а, b], на котором требуется найти максимум. Делим отрезок слева и справа в соответствии с пропорцией золотого сечения и получаем точки л и п.

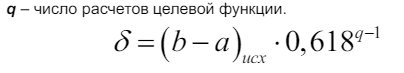


1. Рассчитываем F(л) и F(п).
2. Расстояние между точками л и п не мало и вероятность того, что точка экстремума попадет между ними, достаточно велика.
3. Если F(п) > F(л), нельзя сказать, в какой из трех частей отрезка окажется максимум – может быть в средней части отрезка, или в правой. Но в левой части максимума быть не может. Поэтому его можно отбросить.
4. Переносим левый конец отрезка в точку л, назвав ее а. Получаем новый отрезок [а, b].
5. Но на этом отрезке уже есть точка п, в которой рассчитано значение функции, причем эта точка, отсекавшая от предыдущего большего отрезка справа ~38,2%, отсекает от нового, уменьшенного отрезка справа ~61,8%, т.е. и на новом отрезке она является точкой золотого сечения.
6. На новом этапе расчета называем ее л, на уменьшенном отрезке поставим не две точки для расчета F, а только одну – правую

Получается на каждом этапе расчета, кроме самого первого, мы должны рассчитывать F только в одной точке, что повышает эффективность метода.

Правило останова: 

Эффективность метода:



Метод Фибоначчи

Подобно методу золотого сечения процедура поиска с использованием чисел Фибоначчи требует два вычисления функции на первом шаге, а на каждом последующем - только по одному. Однако, в отличие от метода золотого сечения, в этой процедуре общее число S вычисления функции должно быть выбрано заранее. Кроме того, сокращение интервала неопределенности не остается постоянным, а меняется от шага к шагу: на k-ом шаге длина интервала неопределенности сжимается с коэффициентом .

После (S-1)-го шага точка проведенного вычисления оказывается в середине отрезка локализации. Точка последнего, S-го, шага выбирается на расстоянии δ от середины этого отрезка, где δ - заранее фиксированное малое положительное число (константа различимости)

Алгоритм минимизации функции f(x) с использование, чисел Фибоначчи.

1. По заданной точности рассчитывается вспомогательное число

https://dit.isuct.ru/IVT/sitanov/Literatura/M171/Pages/Glava1_4.files/image018.gif

1. Для полученного значения *N*находится такое число Фибоначчи *FS*, для которого выполняется неравенство:

https://dit.isuct.ru/IVT/sitanov/Literatura/M171/Pages/Glava1_4.files/image020.gif

1. По формуле

https://dit.isuct.ru/IVT/sitanov/Literatura/M171/Pages/Glava1_4.files/image012.gif

определяется длина конечного интервала неопределенности *l.*

1. Вычисляются значения *x1=a+l∙FS-2, x2=b-l∙FS-2.*
2. В зависимости от соотношения *f(x1)* и *f(x2)* выбирается новый интервал локализации минимума.
3. Внутри полученного интервала находится новая точка  *(x1*или *x2),*симметричная к уже имеющейся точке и отстоящая от конца интервала на величину *l∙FS-1-k ,*где *k*- номер шага. В этой точке рассчитывается значение *f(x).* Затем вычисления повторяются, до тех пор, пока *k* не станет равно *S-1*.
4. Полагают *x2=x1+δ,*находят*f(x2).*Если *f(x2)*>*f(x1)*то минимум реализуется на интервале *(a, x1)*, в противном случае - на интервале *(x1, b)*

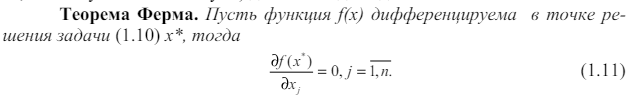
Вопрос 31: Поиск глобального минимума функции одного переменного.

Точка x\*∈X называется точкой глобального (абсолютного) максимума функции f (x) на множестве X, если



Определения глобального минимумов получаются заменой в (1.1) знаков неравенств на противоположные.



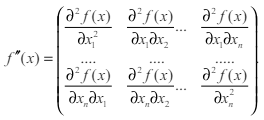


Условие (1.11) или эквивалентное ему  называется условием первого порядка, т.к. использует первые частные производные. Оно является необходимым, но не достаточным, т.е. ему могут удовлетворять и точки, не являющиеся экстремумами.

Так как условия второго порядка для максимумов и минимумов несколько отличаются, то рассмотрим сначала задачу



Будем считать f(x) дважды дифференцируемой и введем матрицу вторых частных производных

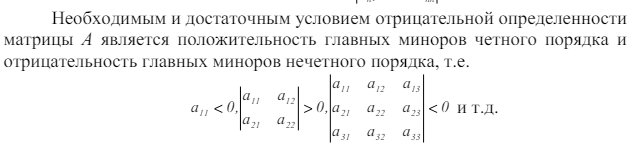




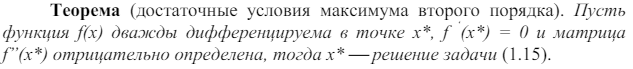
Условием минимума является неотрицательная определенность матрицы f “(x\*)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Часть критерия Сильвестра

Для полуопределенных матриц меняем знак неравенства на не строгий.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*



Вопрос 32: Этапы реализации симплекс-метода.

Симплекс метод является универсальным методом, которым можно решить любую задачу линейного программирования.

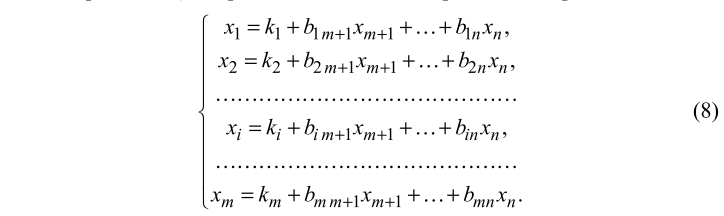
Используя систему ограничений, приведенную к общему виду, т.е. к системе m уравнений с n переменными (m<n), находят ее любое базисное решение, по возможности наиболее простое. Если первое же найденное базисное решение оказалось допустимым, то проверяют его на оптимальность. Если оно не оптимально, то переходят к другому допустимому базисному решению. Симплексный метод гарантирует, что при этом новом решении линейная форма, если не достигнет оптимума, то приблизится к нему. С новым допустимым базисным решением поступают так же, пока не находят решение, которое является оптимальным.

Таким образом, применение симплексного метода распадается на два этапа:

1. нахождение опорного решения системы ограничений;
2. нахождение оптимального решения.

Нахождение опорного решения.

Пусть задана каноническая задача линейного программирования. Выбираем группу m основных переменных, которые позволяют найти исходное базисное решение (можем считать, что основными являются первые m переменных). Выразим эти основные переменные через неосновные:



Этому способу разбиения переменных соответствует базисное решение  Рассмотрим случай, когда это решение является недопустимым. От полученного базисного решения следует сначала перейти к какому-нибудь опорному решению, причем не обязательно, чтобы этот переход осуществлялся сразу, в один шаг.

Если система ограничений не противоречива, то через конечное число шагов будет осуществлен переход к опорному решению.

Так как исходное базисное решение недопустимо (по предположению), среди свободных членов системы ограничений (8) имеется хотя бы один отрицательный (число отрицательных свободных членов этой системы совпадает с числом отрицательных компонент исходного базисного решения). Пусть им является свободный член  i-го уравнения, т.е. основная переменная в соответствующем базисном решении отрицательна.

Это уравнение показывает, что переменная  возрастает при возрастании тех неосновных переменных, коэффициенты которых в этом уравнении положительны. Следовательно, в основные можно переводить те неосновные переменные, которые в уравнении системы (8) с отрицательным свободным членом имеют положительные коэффициенты.

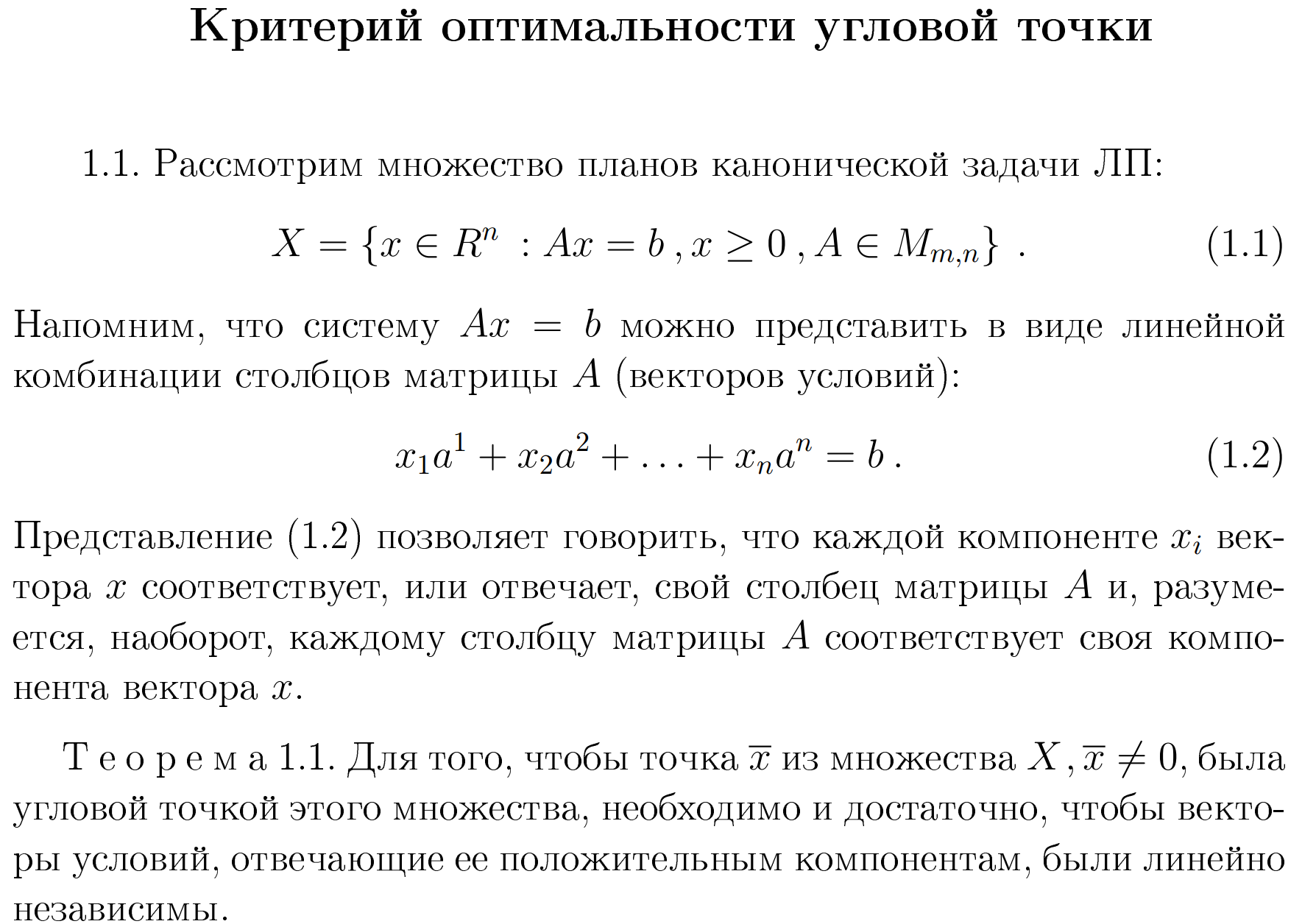
Возможны три случая:

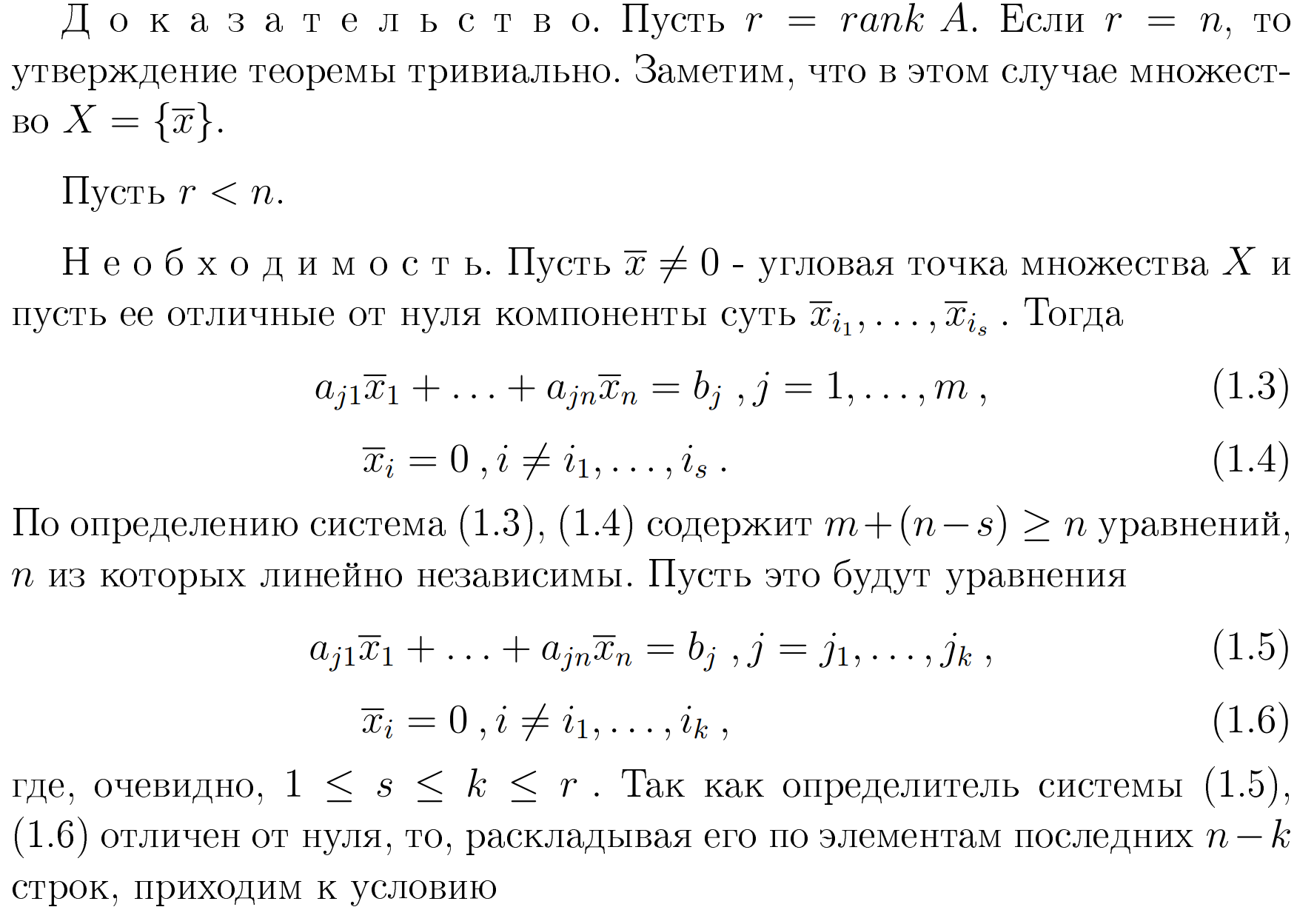
1. В i-м уравнении системы (8) нет неосновных переменных с положительными коэффициентами, т.е. все коэффициенты отрицательны (как и свободный член  . В этом случае данная система ограничений несовместна – она не имеет ни одного допустимого решения. Действительно, вследствие неотрицательности всех переменных, в том числеиз i-го уравнения, в котором свободный член  и все коэффициенты  отрицательны, следует, что переменная  не может принимать неотрицательных значений. Значит, нет и оптимального решения.
2. В i-м уравнении имеется одна переменная , коэффициент при которой положителен. В этом случае именно эта переменная переходит в основные.
3. В i-м уравнении имеется несколько переменных с положительными коэффициентами. В этом случае в основные можно перевести любую из них.

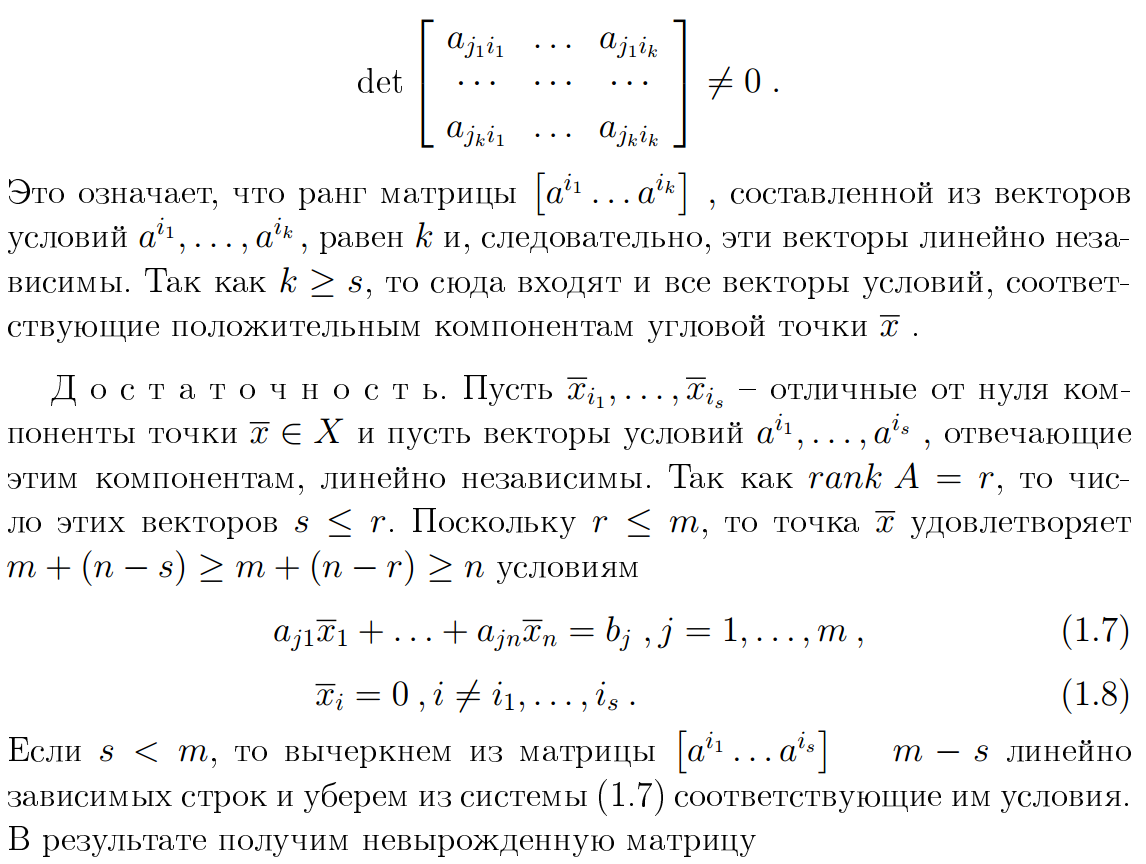
Для того чтобы установить, какая основная переменная должна быть переведена в число неосновных, находят отношения свободных членов к коэффициентам при переменной, переводимой в основные, из всех уравнений, где знаки свободных членов и указанных коэффициентов противоположны, а затем рассматривают абсолютную величину этих отношений и из них выбирают наименьшую (если в некоторых уравнениях знаки свободных членов и коэффициентов совпадают или в каких-то уравнениях переменная, переводимая в основные, отсутствует, то отношение не рассматривают).

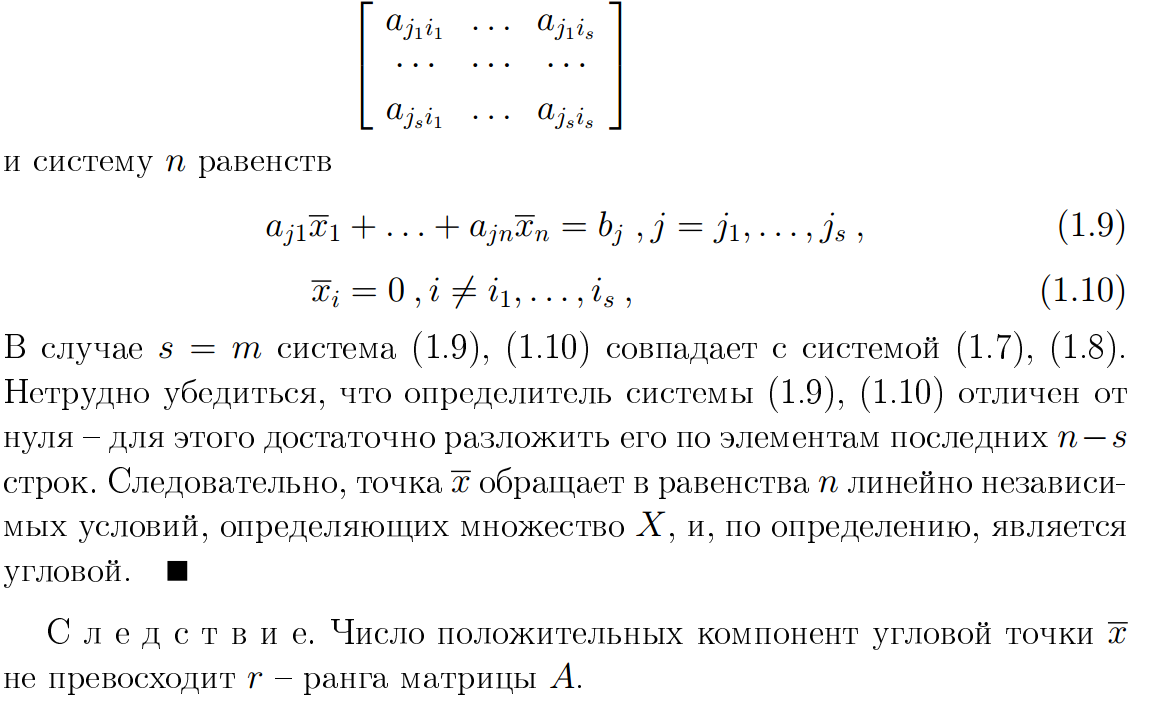
Уравнение, из которого получено наименьшее отношение, выделяют. Выделенное уравнение показывает, какая из основных переменных должна быть переведена в неосновные. Выразив новые основные переменные через неосновные, переходят к следующему базисному решению, которое ближе к опорному. Если оно окажется недопустимым, то к нему следует применить ту же схему еще раз. В результате через конечное число шагов получится опорное решение.

1. Критерий оптимальности угловой точки.

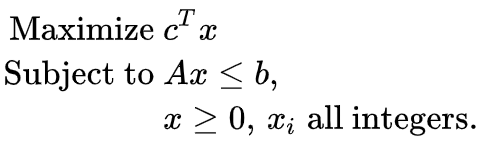








34 Метод отсечения Гомори решения задач целочисленного программирования.



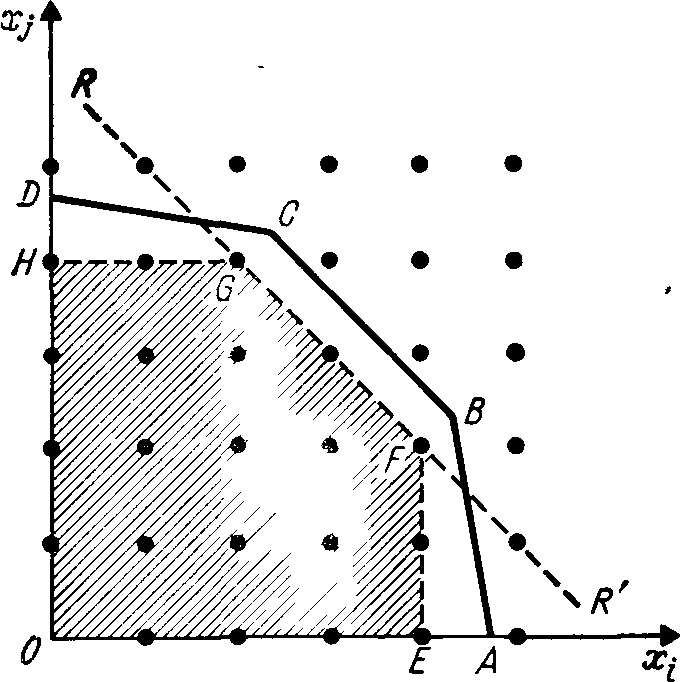
Идея метода Гомори заключается в том, что можно решить задачу обычным симплекс-методом, получив (возможно) нецелочисленное решение, а потом для каждой нецелочисленной переменной добавить "поправочное" ограничение, которое во-первых, сделает эту переменную целочисленной, а во-вторых, минимальным образом повлияет на значение целевой функции:

Суть метода состоит в том, что на каждом шаге к ограничениям соответствующей задачи ЛП добавляется новое ограничение, называемое отсекающей плоскостью. Эта плоскостьдолжна обладать следующими свойствами:

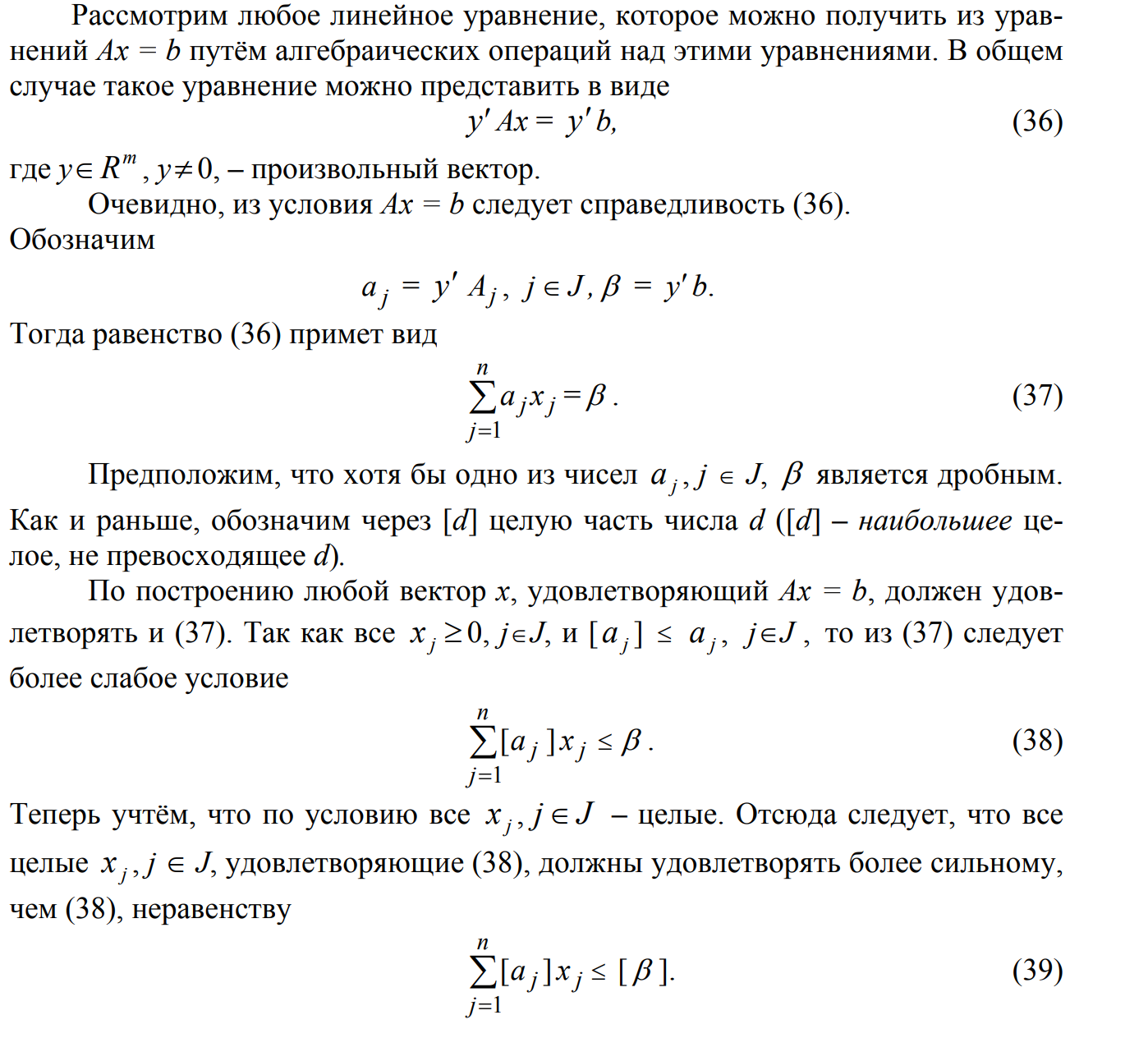
1. отсекать имеющееся в наличии нецелочисленное решение задачи ЛП;
2. не отсекать ни одного целочисленного плана исходной задачи ЦЛП;
3. отсечения должны строиться таким образом, чтобы обеспечить конечность алгоритма, т.е. решение задачи ЦЛП должно строиться за конечное число шагов.

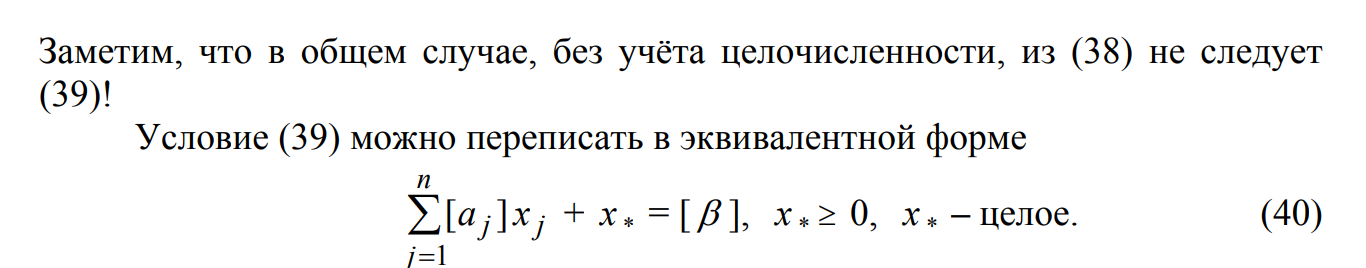
OABCD - исходное решение симплекс-методом

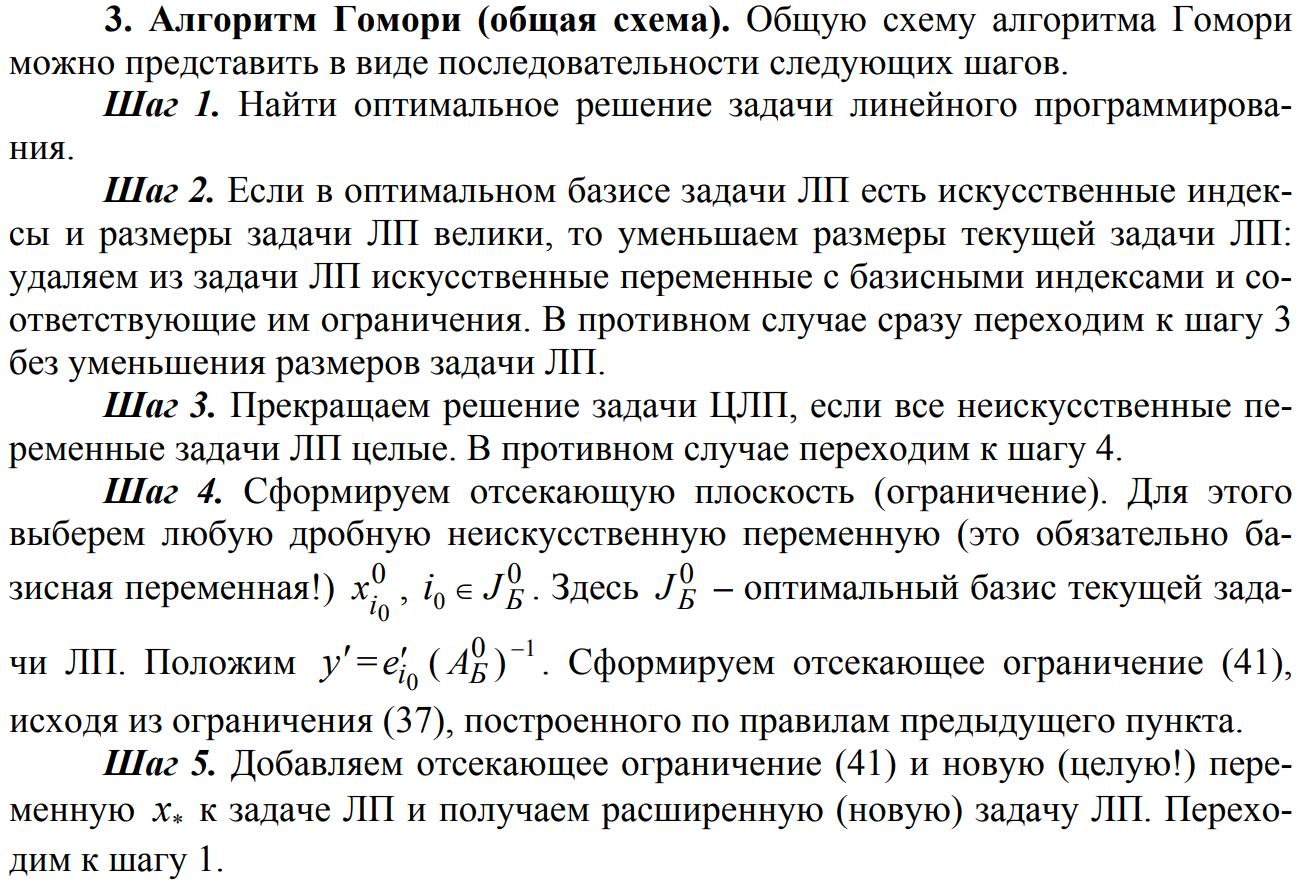
OEFGH - целочисленная область решений



Отсечение:



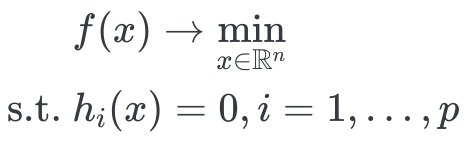




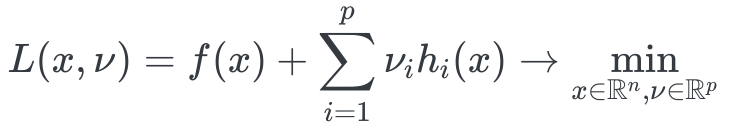
35 Теорема Куна-Таккера.

(Введение о методе множителей Лагранжа)

Пускай дана задача УСЛОВНОЙ оптимизации (на Rn):

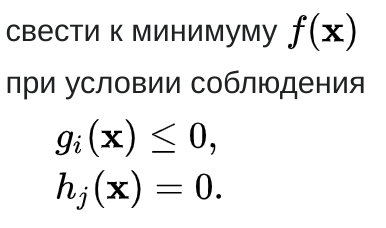


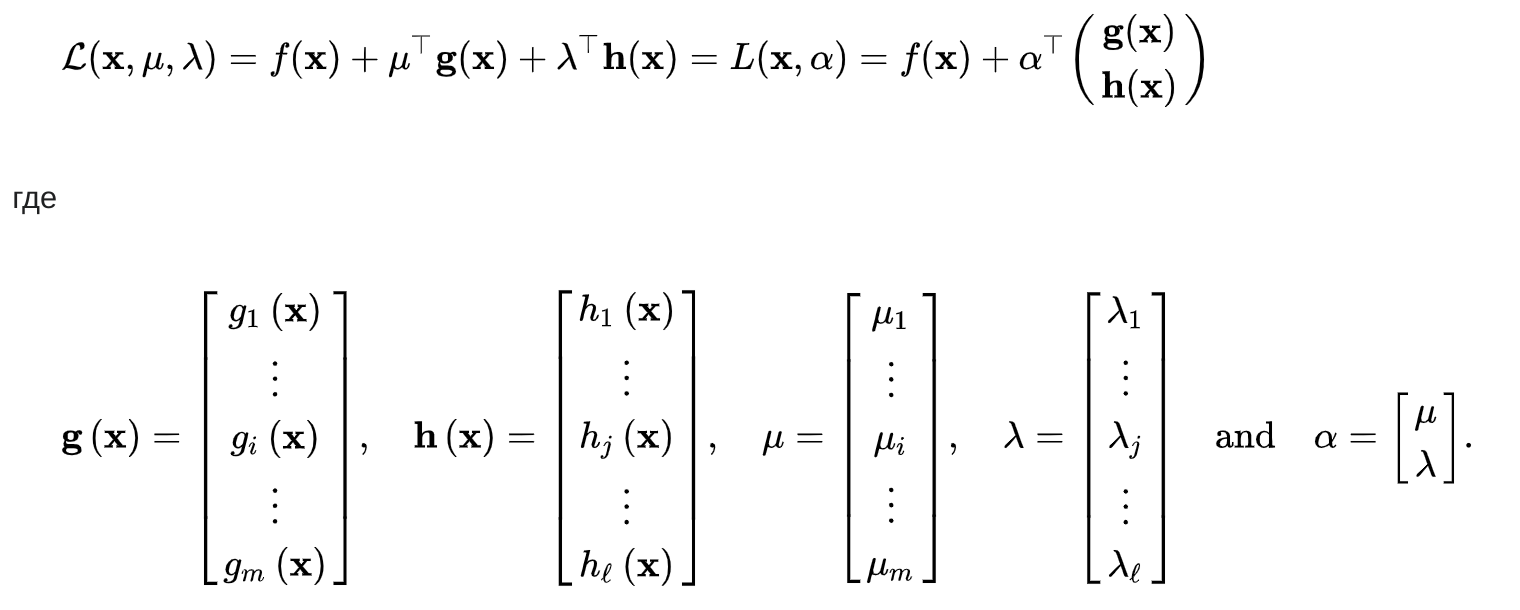
Используя метод множителей Лагранжа можно перейти к БЕЗУСЛОВНОЙ оптимизации, но уже на (Rn,Rp):

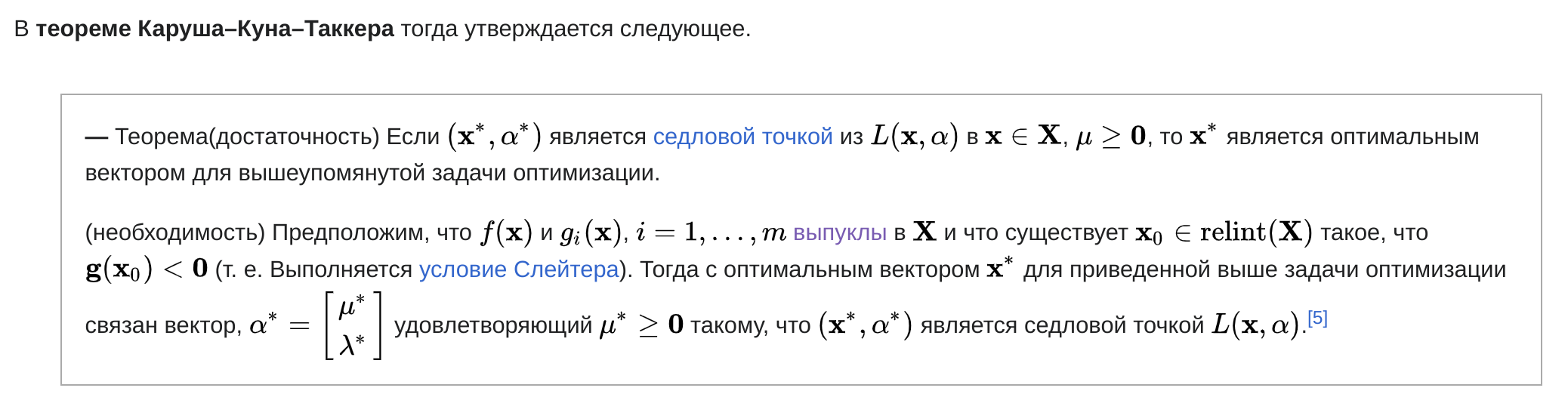


**Теорема Куна-Такера**

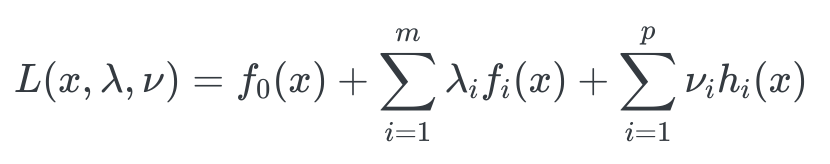
Рассмотрим следующую задачу нелинейной оптимизации в стандартной форме:

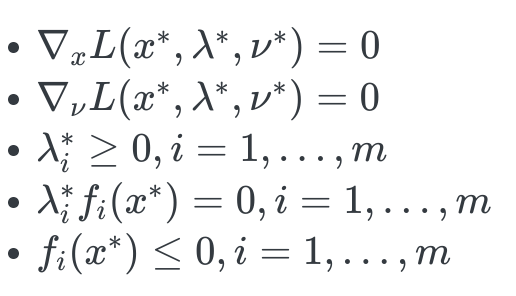


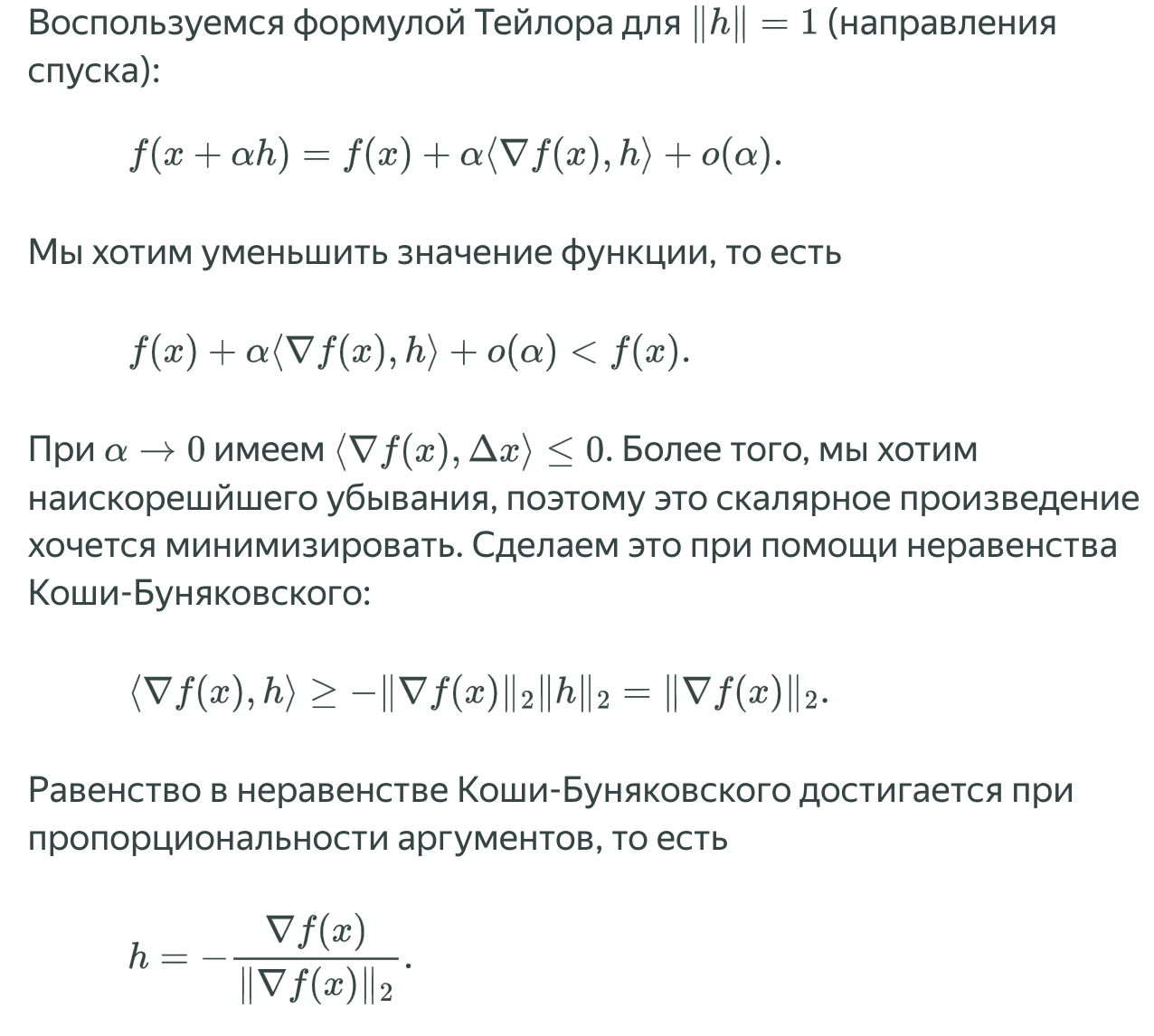


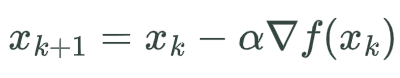


Необходимые условия:

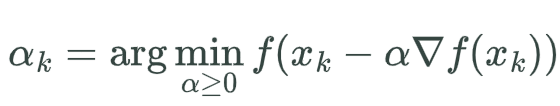




1. Градиентные методы многомерной оптимизации.  
     
   

Тогда алгоритм градиентного спуска:  


Метод наискорейшего спуска:



Одномерная оптимизация является не сильно сложной задачей, поэтому теоретически мы можем её совершать (например, методом бинарного/тернарного поиска или золотого сечения), можно этот шаг также совершать неточно.

37 Метод проекции градиента для задач нелинейного программирования.

38 Метод декомпозиции решения задач оптимального проектирования технической системы (ТС). Согласованность критериев.

39 Построение адаптивных наблюдателей для линейных стационарных систем.

40 Идентификация параметров нелинейных систем, линейно зависящих от параметров.

41 Рекуррентное оценивание параметров.

42 Задача о модальном управлении объектом с неизвестными параметрами и неполной информацией о векторе состояния.

43 Метод наименьших квадратов в задачах совместного оценивания параметров и состояния систем, линейно зависящих от параметров.

44 Компьютерная реализация метода наименьших квадратов.

Вопрос 58: Методы машинного обучения. Их классификация.

Машинное обучение (Machine Learning) — обширный подраздел искусственного интеллекта, изучающий методы построения алгоритмов, способных обучаться.

Разделение методов «по подходам»

* Статистическая классификация
* Классификация на основе сходства

Метрические алгоритмы классификации применяются в тех задачах, где удаётся естественным образом задавать объекты не их признаковыми описаниями, а матрицей попарных расстояний между объектами.

* + метод ближайших соседей;
  + метод парзеновского окна;
  + метод потенциальных функций;
  + метод радиальных базисных функций;
  + отбор эталонных объектов.
* Классификация на основе разделимости

Большая группа методов классификации основана на явном построении разделяющей поверхности в пространстве объектов. Из них чаще всех применяются Линейные классификаторы:

* + линейный дискриминант Фишера;
  + однослойный персептрон;
  + логистическая регрессия;
  + Метод опорных векторов
* Нейронные сети

Нейронные сети основаны на принципе коннективизма — в них соединяется большое количество относительно простых элементов, а обучение сводится к построению оптимальной структуры связей и настройке параметров связей.

* + персептрон;
  + однослойный персептрон
  + многослойный персептрон
  + метод стохастического градиента
  + метод обратного распространения
  + Нейронная сеть Кохонена
  + гибридная сеть встречного распространения
  + сеть радиальных базисных функций
  + оптимальное усечение сети
* Индукция правил (поиск закономерностей)
* Кластеризация
* Регрессия
* Алгоритмические композиции
* Сокращение размерности
* Выбор модели
* Байесовский вывод

Вопрос 59: Задача кластеризации. Методы и алгоритмы ее решения

Формальная постановка задачи кластеризации

Пусть X — множество объектов, Y — множество номеров (имён, меток) кластеров. Задана функция расстояния между объектами \rho(x,x'). Имеется конечная обучающая выборка объектов X^m = \{ x_1, \dots, x_m \} \subset X. Требуется разбить выборку на непересекающиеся подмножества, называемые кластерами, так, чтобы каждый кластер состоял из объектов, близких по метрике \rho а объекты разных кластеров существенно отличались. При этом каждому объекту x_i\in X^m приписывается номер кластера y_i.

Алгоритм кластеризации — это функция a:\, X\to Y которая любому объекту x\in X ставит в соответствие номер кластера y\in Y. Множество Y в некоторых случаях известно заранее, однако чаще ставится задача определить оптимальное число кластеров, с точки зрения того или иного критерия качества кластеризации.

Кластеризация (обучение без учителя) отличается от классификации (обучения с учителем) тем, что метки исходных объектов y_i изначально не заданы, и даже может быть неизвестно само множество Y.

Методы кластеризации.

1. Графовые алгоритмы кластеризации
2. Статистические алгоритмы кластеризации
3. Алгоритм ФорЭл
4. Иерархическая кластеризация или таксономия
5. Нейронная сеть Кохонена
6. Ансамбль кластеризаторов

Вопрос 60: Задача классификации. Методы ее решения.

Классификация — один из разделов машинного обучения, посвященный решению следующей задачи.

Пусть X — множество описаний объектов, Y — конечное множество (имён, меток) классов. Существует неизвестная целевая зависимость — отображение y\*: X -> Y, значения которой известны только на объектах конечной обучающей выборки X^m = \{(x_1,y_1),\dots,(x_m,y_m)\}

Требуется построить алгоритм a:\; X\to Y способный классифицировать произвольный объект x \in X.

Методы решения:

1. Байесовский классификатор — широкий класс алгоритмов классификации, основанный на принципе максимума апостериорной вероятности. Для классифицируемого объекта вычисляются функции правдоподобия каждого из классов, по ним вычисляются апостериорные вероятности классов. Объект относится к тому классу, для которого апостериорная вероятность максимальна.
2. Нейронная сеть — это математическая модель, а также ее программные или аппаратные реализации, построенная в некотором смысле по образу и подобию сетей нервных клеток живого организма.

Нейронные сети — один из наиболее известных и старых методов машинного обучения.

Вопрос 61: Линейная классификация. Регрессионные алгоритмы

Линейный классификатор — алгоритм классификации, основанный на построении линейной разделяющей поверхности. В случае двух классов разделяющей поверхностью является гиперплоскость, которая делит пространство признаков на два полупространства. В случае большего числа классов разделяющая поверхность кусочно-линейна.

Пусть объекты описываются n числовыми признаками f_j:\: X\to\mathbb{R},\; j=1,\ldots,n. Тогда пространство признаковых описаний объектов есть X=\mathbb{R}^n. Пусть Y — конечное множество номеров (имён, меток) классов.

Случай двух классов

Положим Y=\{-1,+1\}. Линейным классификатором называется алгоритм классификации a:\; X\to Y вида

a(x,w) = \mathrm{sign}\left( \sum_{j=1}^n w_j f_j(x) - w_0 \right) = \mathrm{sign}\langle x,w \rangle,

Где w_j — вес j-го признака, w_0 - порог принятия решения, w=(w_0,w_1,\ldots,w_n) — вектор весов, \langle x,w \rangle — скалярное произведение признакового описания объекта на вектор весов. Предполагается, что искусственно введён «константный» нулевой признак: f_{0}(x)=-1

Случай произвольного числа классов

Линейный классификатор определяется выражением

a(x,w) = \mathrm{arg}\max_{y\in Y}\, \sum_{j=0}^n w_{yj} f_j(x) = \mathrm{arg}\max_{y\in Y}\, \langle x,w_y \rangle,

где каждому классу соответствует свой вектор весов w_y=(w_{y0},w_{y1},\ldots,w_{yn})

Регрессионные алгоритмы

Регрессионная модель f(\mathbf{w},\mathbf{x}) — это параметрическое семейство функций, задающее отображение

f:W\times X\longrightarrow Y,

где \mathbf{w}\in W — пространство параметров, \mathbf{x}\in X — пространство свободных переменных, Y — пространство зависимых переменных.

Так как регрессионный анализ предполагает поиск зависимости матожидания случайной величины от свободных переменных E(y|\mathbf{x})=f(\mathbf{x}), то в её состав входит аддитивная случайная величина \varepsilon:

y=f(\mathbf{w},\mathbf{x})+\varepsilon.

Предположение о характере распределения случайной величины \nu называются гипотезой порождения данных. Эта гипотеза играет центральную роль в выборе критерия оценки качества модели и, как следствие, в способе настройки параметров модели.

Модель является настроенной (обученной) когда зафиксированы её параметры, то есть модель задаёт отображение

f:X\longrightarrow Y

для фиксированного значения \bar{\mathbf{w}}.

Линейная регрессия — метод восстановления зависимости между двумя переменными.

Для заданного множества из m пар (x_i, y_i), i=1,\ldots, m начений свободной и зависимой переменной требуется построить зависимость. Назначена линейная модель

y_i= f(\mathbf{w},x_i) + \varepsilon_i

c аддитивной случайной величиной \varepsilon. Переменные x, y принимают значения на числовой прямой \mathbb{R}. Предполагается, что случайная величина распределена нормально с нулевым матожиданием и фиксированной дисперсией \sigma^2_\varepsilon  которая не зависит от переменных x, y. При таких предположениях параметры \mathbf{w} регрессионной модели вычисляются с помощью метода наименьших квадратов.

Нелинейная регрессия — частный случай регрессионного анализа, в котором рассматриваемая регрессионная модель есть функция, зависящая от параметров и от одной или нескольких свободных переменных. Зависимость от параметров предполагается нелинейной.

Задана выборка из m пар (\mathbf{x}_i,y_i). Задана регрессионная модель f(\mathbf{w},\mathbf{x}), которая зависит от параметров \mathbf{w}=(w_1,...,w_W) и свободной переменной x. Требуется найти такие значения параметров, которые доставляли бы минимум сумме квадратов регрессионных остатков

S=\sum_{i=1}^mr_i^2,

где остатки r_i=y_i-f(\mathbf{w},\mathbf{x}_i) для i=1,\ldots,m.

Для нахождения минимума функции S, приравняем к нулю её первые частные производные параметрам \mathbf{w}:

\frac{\partial S}{\partial w_j}=2\sum_i r_i\frac{\partial r_i}{\partial w_j}=0 \ (j=1,\ldots,n). (*)

Так как функция S в общем случае не имеет единственного минимума[[1]](http://www.machinelearning.ru/wiki/index.php?title=%D0%9D%D0%B5%D0%BB%D0%B8%D0%BD%D0%B5%D0%B9%D0%BD%D0%B0%D1%8F_%D1%80%D0%B5%D0%B3%D1%80%D0%B5%D1%81%D1%81%D0%B8%D1%8F" \l "cite_note-one-0), то предлагается назначить начальное значение вектора параметров w_0 и приближаться к оптимальному вектору по шагам:

w_j \approx w_j^{k+1} =w^k_j+\Delta w_j.

Здесь k - номер итерации, \Delta w_j - вектор шага.

На каждом шаге итерации линеаризуем модель с помощью приближения рядом Тейлора относительно параметров \mathbf{w}^k

Здесь элемент матрицы Якоби J_{ij} - функция параметра w_j; значение свободной переменной \mathbf{x}_i фиксировано. В терминах линеаризованной модели

\frac{\partial r_i}{\partial w_j}=-J_{ij}

и регрессионные остатки определены как

r_i=\Delta y_i- \sum_{j=1}^{n} J_{ij}\Delta w_j; \ \Delta y_i=y_i- f(x_i,\mathbf{w}^k).

Подставляя последнее выражение в выражение (\*), получаем

-2\sum_{i=1}^{m}J_{ij} \left( \Delta y_i-\sum_{s=1}^{n} J_{is}\Delta w_s \right)=0.

Преобразуя, получаем систему из n линейных уравнений, которые называются нормальным уравнением

\sum_{i=1}^{m}\sum_{s=1}^{n} J_{ij}J_{is}\Delta w_s=\sum_{i=1}^{m} J_{ij}\Delta y_i (j=1,n).

Запишем нормальное уравнение в матричном обозначении как

\mathbf{\left(J^TJ\right)\Delta \mathbf{w}=J^T\Delta y}.

В том случае, когда критерий оптимальности регрессионой модели задан как взвешенная сумма квадратов остатков

S=\sum_{i=1}^{m}W_{ii}r_i^2,

нормальное уравнение будет иметь вид

\mathbf{\left(J^TWJ\right)\Delta \mathbf{w}=J^TW\Delta y}.

Для нахождения оптимальных параметров нелинейных регрессионных моделей используются метод сопряжённых градиентов, метод Ньютона-Гаусса или алгоритм Левенберга-Марквардта.

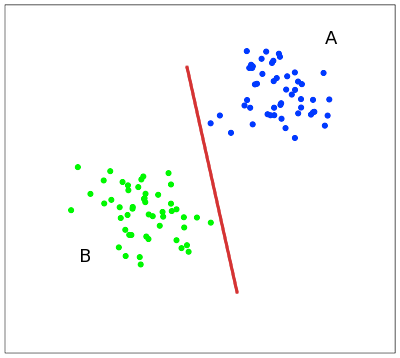
ВОПРОС62: Метод опорных векторов

Данный метод изначально относится к бинарным классификаторам, хотя существуют способы заставить его работать и для задач мультиклассификации.

Идею метода удобно проиллюстрировать на следующем простом примере: даны точки на плоскости, разбитые на два класса (рис. 1). Проведем линию, разделяющую эти два класса (красная линия на рис. 1). Далее, все новые точки (не из обучающей выборки) автоматически классифицируются следующим образом:

точка выше прямой попадает в класс A,

точка ниже прямой — в класс B.



Такую прямую назовем *разделяющей* прямой, а в пространствах разменостью больше трех назовем гиперплоскостью. С точки зрения точности классификации лучше всего выбрать прямую, расстояние от которой до каждого класса максимально.

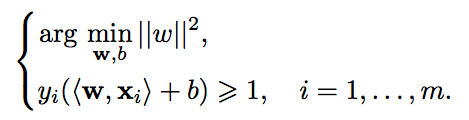
Формализация.

Пусть имеется обучающая выборка: https://habrastorage.org/r/w1560/storage/habraeffect/ec/0f/ec0fd2147020136102f199d847315336.png

Метод опорных векторов строит классифицирующую функцию *F* в виде https://habrastorage.org/r/w1560/storage/habraeffect/9e/39/9e39396ca18b921d9afbef6d92607ddb.png

где https://habrastorage.org/r/w1560/storage/habraeffect/bc/aa/bcaa9f6be3acdb74579d883fa63c21f3.png — скалярное произведение, ***w*** — нормальный вектор к разделяющей гиперплоскости,*b* — вспомогательный параметр. Те объекты, для которых *F(****x****) = 1* попадают в один класс, а объекты с*F(****x****) = -1* — в другой. Выбор именно такой функции неслучаен: любая гиперплоскость может быть задана в виде https://habrastorage.org/r/w1560/storage/habraeffect/bb/77/bb77637fa821ab46507330ed645ceb33.png для некоторых ***w*** и *b.*

Далее, мы хотим выбрать такие ***w*** и *b* которые максимизируют расстояние до каждого класса. Можно подсчитать, что данное расстояние равно https://habrastorage.org/r/w1560/storage/habraeffect/31/86/3186bd99eb78eadf2af35359b431c58e.png. Проблема нахождения максимума https://habrastorage.org/r/w1560/storage/habraeffect/31/86/3186bd99eb78eadf2af35359b431c58e.png эквивалентна проблеме нахождения минимума https://habrastorage.org/r/w1560/storage/habraeffect/05/47/0547d2ad77f3163ca1641d382a173ad2.png. Запишем все это в виде задачи оптимизации:



которая является стандартной задачей квадратичного программирования и решается с помощью множителей Лагранжа.

Вопрос63: Метод деревьев решений

Дерево решений — метод представления решающих правил в определенной иерархии, включающей в себя элементы двух типов — узлов (node) и листьев (leaf). Узлы включают в себя решающие правила и производят проверку примеров на соответствие выбранного атрибута обучающего множества.

Рассмотрим бинарное дерево, в котором:

* каждой внутренней вершине v приписана функция (или предикат) 
* каждой листовой вершине v приписан прогноз  (в случае с классификацией листу также может быть приписан вектор вероятностей).

Рассмотрим теперь алгоритм a(x), который стартует из корневой вершины и вычисляет значение функции . Если оно равно нулю, то алгоритм переходит в левую дочернюю вершину, иначе в правую, вычисляет значение предиката в новой вершине и делает переход или влево, или вправо. Процесс продолжается, пока не будет достигнута листовая вершина; алгоритм возвращает тот класс, который приписан этой вершине. Такой алгоритм называется бинарным решающим деревом.

Опишем базовый жадный алгоритм построения бинарного решающего дерева. Начнем со всей обучающей выборки X и найдем наилучшее ее разбиение на две части  с точки зрения заранее заданного функционала качества . Найдя наилучшие значения j и t, создадим корневую вершину дерева, поставив ей в соответствие предикат . Объекты разобьются на две части — одни попадут в левое поддерево, другие в правое. Для каждой из этих подвыборок рекурсивно повторим процедуру, построив дочерние вершины для корневой, и так далее. В каждой вершине мы проверяем, не выполнилось ли некоторое условие останова — и если выполнилось, то прекращаем рекурсию и объявляем эту вершину листом. Когда дерево построено, каждому листу ставится в соответствие ответ. В случае с классификацией это может быть класс, к которому относится больше всего объектов в листе, или вектор вероятностей. Для регрессии это может быть среднее значение, медиана или другая функция от целевых переменных объектов в листе. Выбор конкретной функции зависит от функционала качества в исходной задаче.

Вопрос64: Вероятностные методы. Наивный байесовский классификатор

Наивный байесовский классификатор — специальный частный случай байесовского классификатора, основанный на дополнительном предположении, что объекты x\in X описываются n статистически независимыми признаками:

x \equiv \bigl( \xi_1,\ldots,\xi_n\bigr) \equiv \bigl( f_1(x),\ldots,f_n(x) \bigr)

Предположение о независимости означает, что функции правдоподобия классов представимы в виде

p_y(x) = p_{y1}(\xi_1) \cdot \ldots \cdot p_{yn}(\xi_n)

Предположение о независимости существенно упрощает задачу, так как оценить n одномерных плотностей гораздо легче, чем одну n-мерную плотность. К сожалению, оно крайне редко выполняется на практике, отсюда и название метода.

*Наивный байесовский классификатор* может быть, как параметрическим, так и непараметрическим, в зависимости от того, каким методом восстанавливаются одномерные плотности.

Основные преимущества *наивного байесовского классификатора* — простота реализации и низкие вычислительные затраты при обучении и классификации. В тех редких случаях, когда признаки действительно независимы (или почти независимы), наивный байесовский классификатор (почти) оптимален.

Основной его недостаток — относительно низкое качество классификации в большинстве реальных задач.

Чаще всего он используется либо как примитивный эталон для сравнения различных моделей алгоритмов, либо как элементарный строительный блок в алгоритмических композициях.

ВОПРОС65: Ансамблевые методы. Методы случайного леса. Метод градиентного бустинга.

Метод машинного обучения, где несколько моделей обучаются для решения одной и той же проблемы и объединяются для получения лучших результатов называется ансамблевым методом. Основная предпосылка заключается в том, что результат работы нескольких моделей будет более точен, чем результат только одной модели.

Когда говорится об ансамблях, то вводится понятие слабого ученика(обычные модели вроде линейной регрессии или дерева решений). Множество слабых учеников являются строительными блоками для более сложных моделей. Объединение слабых учеников для улучшения качества модели, уменьшения смещения или разброса, называется сильным учеником.

Виды ансамблевых методов

Наиболее популярными ансамблевыми методами являются: стекинг, бэггинг, бустинг.

Стекинг. Используется несколько разнородных слабых учеников. Их обучают и объединяют для построения прогноза, основанного на результатах различных слабых моделей.

Бэггинг. В этом случае однородные модели обучают на разных наборах данных и объединяют. Получают прогноз путём усреднения. Если использовать в качестве слабого ученика деревья решений, то получится случайный лес RandomForestClassifier / RandomForestRegressor.

Бустинг. При использовании данного метода несколько однородных моделей последовательно обучаются, исправляя ошибки друг друга.

Методы случайного леса делятся на два вида RandomForestClassifier и RandomForestRegressor.

RandomForestClassifier - это метаоценщик, который соответствует ряду классификаторов дерева решений для различных подвыборок набора данных и использует усреднение для повышения точности прогнозирования и контроля переподбора. Деревья в лесу используют лучшую стратегию разделения.

RandomForestRegressor - это метаоценщик, который подбирает несколько регрессоров дерева решений для различных подвыборок набора данных и использует усреднение для повышения точности прогнозирования и контроля переобучения. Деревья в лесу используют лучшую стратегию разделения

Градиентный бустинг

Градиентный бустинг обучает слабые модели последовательно, исправляя ошибки предыдущих. Результатом градиентного бустинга также является средневзвешенная сумма результатов моделей. Адаптивный бустинг использует итеративный метод оптимизации. Градиентный бустинг оптимизируется с помощью градиентного спуска.

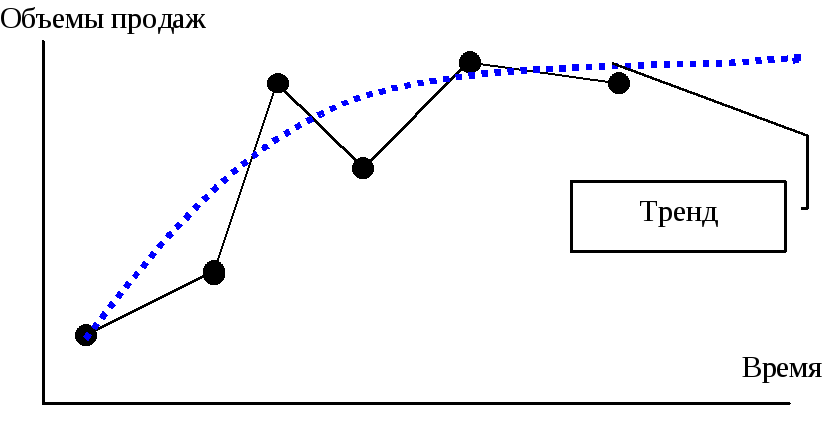
Таким образом градиентный бустинг - обобщение адаптивного бустинга для дифференцируемых функций.

Вопрос66: Методы прогнозирования. Временные ряды

Временной ряд — собранный в разные моменты времени статистический материал о значении каких-либо параметров (в простейшем случае одного) исследуемого процесса. Каждая единица статистического материала называется измерением или отсчётом, также допустимо называть его уровнем на указанный с ним момент времени. Во временном ряде для каждого отсчёта должно быть указано время измерения или номер измерения по порядку. Временной ряд существенно отличается от простой выборки данных, так как при анализе учитывается взаимосвязь измерений со временем, а не только статистическое разнообразие и статистические характеристики выборки.

Существует несколько методов прогнозирования временных рядов, включая методы экстраполяции, методы машинного обучения (регрессия, нейронные сети) и прогнозирование с помощью SARIMA.

Методы экстраполяции - это методы, которые используют значения предыдущих периодов для прогнозирования будущих значений ряда. Методы экстраполяции включают метод скользящего среднего, экспоненциальное сглаживание и метод Хольта-Винтерса. Метод скользящего среднего заключается в вычислении среднего значения ряда за определенный период, а затем использовании этого значения для прогнозирования будущих значений.



Экспоненциальное сглаживание использует экспоненциальную функцию для прогнозирования будущих значений, а метод Хольта-Винтерса добавляет тренд и сезонность к модели экспоненциального сглаживания.

Методы машинного обучения - это методы прогнозирования, которые используют данные прошлых значений ряда в качестве входных данных для модели обучения. Регрессия - это метод машинного обучения, который моделирует зависимость между независимыми (входными) и зависимыми (выходными) переменными. В случае прогнозирования временных рядов, независимыми переменными будут значения ряда в прошлых периодах, а зависимой переменной - будущие значения ряда. Нейронные сети - это более сложные методы машинного обучения, которые используют алгоритмы обработки информации, подобные тем, которые используют мозг. Нейронные сети могут использоваться для прогнозирования временных рядов и обычно имеют более высокую точность, чем методы экстраполяции.

Прогнозирование с помощью SARIMA - это метод прогнозирования, который использует модель SARIMA (Seasonal AutoRegressive Integrated Moving Average). Эта модель объединяет в себе методы авторегрессии, интегрированного скользящего среднего и сезонности, что позволяет прогнозировать будущие значения ряда на основе его предыдущих значений с учетом сезонных паттернов. SARIMA широко используется в экономике и финансах для прогнозирования цен на акции, индексы фондового рынка и т.д.

Вопрос67: Нейронные сети.

Нейро́нная сеть (также иску́сственная нейро́нная сеть, ИНС, или просто нейросе́ть) — математическая модель, а также её программное или аппаратное воплощение, построенная по принципу организации биологических нейронных сетей — сетей нервных клеток живого организма. В отличие от нейросети животного, которая передаёт сигнал от мозга к другим органам и полностью регулирует жизнедеятельность организма, компьютерная нейросеть учится решать только ту задачу, которую ей ставит человек.

Виды нейронных сетей

Классификация нейронных сетей основана на задачах, с которыми они работают:

* многослойные нейронные сети, или перцептроны, обрабатывают числовые данные;
* свёрточные нейронные сети работают с изображениями;
* рекуррентные нейронные сети собирают и обрабатывают информацию, которая меняется с течением времени;
* генеративные нейронные сети создают контент — тексты, изображения.

Принцип работы нейронной сети

Весь процесс можно разделить на шесть этапов.

1. Постановка задачи

С этого начинается работа над построением нейронной сети.

2. Сбор исходных данных

Для работы нейросети нужна информация, на основе которой она будет учиться искать решение. Данные должны быть качественными, потому что сеть отчасти похожа на ребёнка: подай плохой пример, и она будет им руководствоваться, скажи плохое слово, и будет его повторять.

3. Анализ данных

Это нужно, чтобы выяснить, нет ли скрытых зависимостей или некорректных данных.

4. Обучение нейронной сети

Нейросети показывают часть данных, чтобы она поняла взаимосвязь между ними, и периодически проверяют качество работы. Обычно тренируют несколько нейронных сетей, выбирают наиболее качественную из них и продолжают работать с ней.

5. Мониторинг нейросети

Смотрят, насколько хорошо модель работает на реальных данных. Как только она начинает плохо справляться с поставленной задачей, её дообучают — показывают несколько примеров новых данных до тех пор, пока она не исправится.

Чтобы не упустить момент, когда нейросеть начинает ошибаться, используют метод human-in-the-loop, в ходе которого человек решает ту же самую задачу на основе тех же данных, а затем специалист по Data Science сравнивает результаты. Если выясняется, что задача решена неверно, нейронную сеть снова дообучают.

6. Дообучение нейросети

Постоянное обучение — основа работы любой нейронной сети. Процесс проверок и дообучения идёт по кругу до тех пор, пока применение нейросети не утратит смысл.

Нейросети не обязательно обучать с нуля — достаточно «подтянуть» их знания по нужным параметрам. Это называется обучением с переносом опыта (англ. transfer learning).

Вопрос68: Глубокие нейронные сети: Свёрточные нейронные сети (CNN), Рекуррентные нейронные сети (RNN).

Глубокое обучение — совокупность методов машинного обучения, основанных на обучении представлениям, а не специализированных алгоритмах под конкретные задачи. Например, в компьютерном зрении, машинном переводе, распознавании речи, причём качество решения во многих случаях теперь сопоставимо, а в некоторых превосходит эффективность человека.

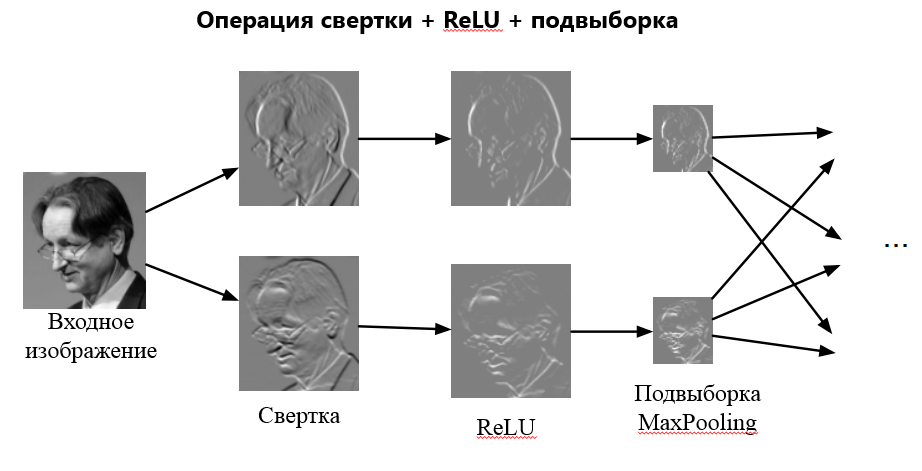
Свёрточная нейронная сеть (convolutional neural network, CNN) — специальная архитектура искусственных нейронных сетей, нацеленная на эффективное распознавание образов, входит в состав технологий глубокого обучения (deep learning).

Структура сверточной нейронной сети  
СНС состоит из разных видов слоев: сверточные (convolutional) слои, субдискретизирующие (subsampling, подвыборка) слои и слои «обычной» нейронной сети – персептрона.



Первые два типа слоев (convolutional, subsampling), чередуясь между собой, формируют входной вектор признаков для многослойного персептрона.

На рисунке ниже продемонстрирована визуализация свертки и подвыборки:



Рекуррентная нейронная сеть (Recurrent Neural Network, RNN) — популярный вид нейронных сетей, используемых в обработке естественного языка а так же временных рядов. Рекуррентная нейросеть оценивает произвольные предположения на основе того, насколько часто они встречались. Это дает меру корректности, что позволяет использовать такие модели для перевода текстов или анализа временных рядов. Кроме того эти модели способны генерировать тексты.

Идея RNN заключается в последовательном использовании информации. В традиционных нейронных сетях подразумевается, что все входы и выходы независимы. Но для многих задач это не подходит. Если вы хотите предсказать следующее слово в предложении, лучше учитывать предшествующие ему слова. RNN называются рекуррентными, потому что они выполняют одну и ту же задачу для каждого элемента последовательности, причем выход зависит от предыдущих вычислений. Еще одна интерпретация RNN: это сети, у которых есть «память», которая учитывает предшествующую информацию. Теоретически рекуррентная нейронная сеть может использовать информацию в произвольно длинных последовательностях, но на практике она ограничена лишь несколькими шагами



На диаграмме выше показано, что RNN разворачивается в полную сеть. Разверткой мы просто выписываем сеть для полной последовательности. Например, если последовательность представляет собой предложение из 5 слов, развертка будет состоять из 5 слоев, по слою на каждое слово. Формулы, задающие вычисления в RNN следующие:

* x\_t — вход на временном шаге t. Например x\_1 может быть вектором с одним горячим состоянием (one-hot vector), соответствующим второму слову предложения.
* s\_t — это скрытое состояние на шаге t. Это «память» сети. s\_t зависит, как функция, от предыдущих состояний и текущего входа x\_t: s\_t=f(Ux\_t+Ws\_{t-1}). Функция f обычно нелинейная, например tanh или ReLU. s\_{-1}, которое требуется для вычисление первого скрытого состояния, обычно инициализируется нулем (нулевым вектором).
* o\_t — выход на шаге t. Например, если мы хотим предсказать слово в предложении, выход может быть вектором вероятностей в нашем словаре. o\_t = softmax(Vs\_t)

Несколько заметок:

* Можно интерпретировать s\_t как память сети. s\_t содержит информацию о том, что произошло на предыдущих шагах времени. Выход o\_t вычисляется исключительно на основе «памяти» s\_t. На практике все немного сложнее: s\_t не может содержать информацию слишком большого количества предшествующих шагов;
* В отличие от традиционной глубокой нейронной сети, которая использует разные параметры на каждом слое, RNN имеет одинаковые (U, V, W) на всех этапах. Это отражает тот факт, что мы выполняем одну и ту же задачу на каждом шаге, используя только разные входы. Это значительно уменьшает общее количество параметров, которые нам нужно подобрать;
* Диаграмма выше имеет выходы на каждом шаге, но, в зависимости от задачи, они могут не понадобиться. Например при определении эмоциональной окраски предложения, целесообразно заботиться только о конечном результате, а не о окраске после каждого слова. Аналогично, нам может не потребоваться ввод данных на каждом шаге. Основной особенностью RNN является скрытое состояние, которое содержит некоторую информацию о последовательности.

Где используют рекуррентные нейросети

* Языковое моделирование и генерация текстов

Учитывая последовательность слов, мы хотим предсказать вероятность каждого слова (в словаре)

* Машинный перевод

Машинный перевод похож на языковое моделирование, поскольку вектор входных параметров представляет собой последовательность слов на исходном языке, хотим получить последовательность слов на целевом языке

* Распознавание речи

По входной последовательности акустических сигналов от звуковой волны, мы можем предсказать последовательность фонетических сегментов вместе со своими вероятностями.

* Генерация описания изображений

Вместе со сверточными нейронными сетями RNN использовались как часть модели генерации описаний неразмеченных изображений. Комбинированная модель совмещает сгенерированные слова с признаками, найденными на изображениях.