Министерство образования Республики Беларусь

Учреждение образования
«Гомельский государственный технический
университет имени П. О. Сухого»

Кафедра «Машины и технология литейного производства»

В. А. Жаранов

МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ ТЕХНОЛОГИЧЕСКИХ ПРОЦЕССОВ

КУРС ЛЕКЦИЙ
по одноименной дисциплине
для студентов специальности 1-36 02 01
«Машины и технология литейного производства»
дневной и заочной форм обучения

Гомель 2009

УДК 621.745(075.8)
ББК 34.61я73
Ж34

Рекомендовано научно-методическим советом механико-технологического факультета ГГТУ им. П. О. Сухого (*протокол № 9 от 15.09.2009 г.*)

Рецензент: зав. каф. «Обработка материалов давлением» ГГТУ им. П. О. Сухого д-р техн. наук, проф. *М. Н. Верещагин*

Жаранов, В. А.

Ж34 Математическое моделирование технологических процессов : курс лекций по одноим. дисциплине для студентов специальности 1-36 02 01 «Машины и технология литейного производства» днев. и заоч. форм обучения / В. А. Жаранов. – Гомель : ГГТУ им. П. О. Сухого, 2009. – 120 с. – Систем. требования: PC не ниже Intel Celeron 300 МГц ; 32 Mb RAM ; свободное место на HDD 16 Mb ; Windows 98 и выше ; Adobe Acrobat Reader. – Режим доступа: http://lib.gstu.local. – Загл. с титул. экрана.

Рассмотрены основные методы математического моделирования технических систем и процессов. Представлена информация о методах оптимизации и численных методах. Изложены особенности исследования и оптимизации параметров технологий литья.

Для студентов специальности 1-36 02 01 «Машины и технология литейного производства» дневной и заочной форм обучения.

УДК 621.745(075.8)
ББК 34.61я73

ВВЕДЕНИЕ.

Моделирование представляет собой метод исследования свойств одного объекта посредством изучения свойств другого объекта, более удобного для исследования и находящегося в определенном соответствии с первым объектом.

Методы моделирования применяются практически во всех областях деятельности человека, при решении научно-технических задач, для изучения социальных, экономических, медицинских, военных или экологических проблем. В любой сфере деятельности человека моделирование находит свое применение.

Общеизвестно, что изучение аэродинамических свойств самолета производится, кроме всего прочего, в аэродинамической трубе, куда помещается сначала уменьшенная копия самолета, а на заключительном этапе исследований и сам самолет. При воздействии на объект воздушного потока проверяется, как на разных скоростях полета воздух обтекает самолет. Таким образом, устанавливают - оптимальна ли форма самолета, и надо ли ее дорабатывать.

Другое применение аэродинамических труб, это продувка автомобилей (им желательно придавать более обтекаемую форму, чем добиваются уменьшения лобового сопротивления и, следовательно, уменьшения расхода топлива, т.е. повышения экономичности эксплуатации), продувка макетов кораблей также позволяет судить об их ветровых качествах, хотя скорость корабля намного меньше, чем у самолета или автомобиля, но и ветер на водной поверхности достигает большего значения и может либо сильно замедлить скорость хода (придется увеличивать расход топлива), либо вообще перевернуть корабль.

Еще одно интересное применение аэродинамической трубы - продувание макета здания, с целью проверки на ветроустойчивость. Примером служит история, произошедшая в г. Бостон (США), где после строительства нового 60-и этажного здания пришлось сменить все окна, что обошлось в 7 млн. долларов. Как исправить просчет проектировщиков выявили после испытания макета здания в аэродинамической трубе, изучив особенности ветровых нагрузок на стены здания.

Исторически первыми моделями как заместителями некоторых объектов были, видимо, символические условные модели. Это языковые знаки, которые в ходе развития составили разговорный язык.

Применение символических условных моделей другого типа связано, вероятно, с возникновением обмена: сначала предметы раскладывались в два ряда, друг напротив друга, чем и добивались однозначного соответствия, потом было установлено, что соответствия объектов одного рода объектам второго рода можно добиться сравнивая их с объектами третьего рода, сначала с естественными объектами - пальцы рук и ног, затем с искусственными - специально изготовленные палочки. Эти первые логические условные модели постепенно привели к формированию понятию числа.

Следующий этап развития логического моделирования - возникновение знаковых числовых обозначений.

В глубокой древности возник и получил развитие метод распространения свойств одних объектов на другие, который теперь называется умозаключением по аналогии.

Дальнейшее развитие логических знаковых моделей связано с возникновением письменности и математической символики, а это относится примерно к 2000г. до н. э., время расцвета цивилизаций Египта и Вавилона. Вавилоняне пользовались таким важным для моделирования понятием, как подобие в форме элементарного геометрического подобия прямоугольных треугольников.

Развитие моделирование получает в Древней Греции в V - III вв. до н. э. В Греции была создана геометрическая модель Солнечной системы, греческий врач Гиппократ для изучения глаза человека пользовался глазом быка, его физической аналогичной моделью, математик Евклид построил учение о геометрическом подобии.

Вопросы подобия в связи с созданием различных конструкций и их моделированием часто возникают в XVI - XVII вв. О том, что подобию стали уделять много внимания в XVII в. пишет Г. Галилей в своем сочинении «Разговоры о двух новых науках». Например, при постройке в Венеции галеры с увеличенными размерами подпорки с сечениями, выбранными исходя из геометрического подобия, оказались недостаточно прочными, и размеры их пришлось корректировать на основе физических соотношений. Галилей констатировал, что «прочность подобных тел не сохраняет того же отношения, которое существует между величиной тел».

Первые строгие научные формулировки условий подобия и уточнения этого понятия были даны применительно к механическому движению в конце XVII в. И. Ньютоном в работе «Математические

начала натуральной философии». В работе рассматриваются движения материальных тел и устанавливаются законы их подобия. Основы современного учения о подобии заложили, сформулированные И. Ньютоном, прямая теорема подобия и основные положения подобия, указав свойства подобных механических систем и критерии, характеризующие движения систем, подобие которых обеспечено. И. Ньютон открыл пути применения подобия и моделирования для обоснования теоретических положений. Им построена наглядная механическая модель для объяснения световых явлений (корпускулярная теория света), математическая модель для объяснения явления тяготения и мн. др.

Работы И. Ньютона по теории подобия и моделирования долгое время не получали развития, хотя в начале XVIII в. во Франции и других странах проводились многочисленные опыты на моделях арок и проверялись различные гипотезы работы их свода.

Одним из первых теоретически обоснованно применил статическое подобие И.П. Кулибин при разработке проекта арочного моста, пролетом 300 м. Исследования он проводил на деревянных моделях в 1/10 натуральной величины. В них было впервые учтено, что увеличение линейных размеров в k раз меняет собственный вес в k^3 раз, а площади поперечных сечений элементов - в k^2 раз. И.П. Кулибин установил, что обеспечение подобия влияния собственного веса в модели возможно при некоторой дополнительной нагрузке. Предложенный метод моделирования собственного веса конструкции соответствует современному способу «догрузки» моделей в центрифугах.

В 1822 г. появились работа Ж. Фурье «Аналитическая теория теплопроводности», в которой было показано, что члены уравнений, описывающих физические явления, всегда имеют одинаковую размерность, это свойство получило название правила Фурье или правила размерной однородности уравнений математической физики. В 1848 г. Ж.Л.Ф. Бертран, пользуясь методом подобных преобразований, установил наиболее общие свойства подобных механических движений и указал способы осуществления подобия сложного механического движения, четко сформулировав положение о наличии критериев подобия. Вскоре появился ряд работ, посвященных приложению теории подобия к различным механическим явлениям. Например, законы звуковых явлений в геометрически подобных телах из уравнения движения упругих тел;

условия подобия гидродинамических явлений. Появились работы в области строительной механики, в области упругости.

Однако, практическое применение теории подобия и моделирования, зачастую встречало серьезные препятствия, трагическим примером чему служит история с английским броненосцем «Кэптен». Этот корабль построили в 1870 году. В то же время английские ученые-кораблестроители Фруд и Рид создали теорию моделирования кораблей, исследование модели броненосца показало, что он должен опрокинуться даже при небольшом волнении. Специалисты Адмиралтейства не придали значения опытам ученых с «игрушечной» моделью, в результате при выходе в море «Кэптен» перевернулся, и 523 моряка погибли.

Примером удачного применения методов моделирования является их применение Д.И. Журавским при сооружении железнодорожных мостов. Ранее для определения размеров составных частей ферм мостов применялись упрощенные приемы и все раскосы, и тяжи каждой фермы моста делались одного и того же размера. Выводы о том, что их нагрузки неодинаковы, сначала казались неправдоподобными и были проверены на модели из металлической проволоки. На этой модели оказалось возможным, проводя смычком от скрипки по проволокам, по высоте тона получаемого звука определить степень натяжения проволок, т.е. элементов крепления моста.

Развитие учения о подобии долгое время шло путем определения частных условий подобия для явлений только определенной физической природы. Наконец, в 1909 - 1914 гг. В результате работ Н.Е. Жуковского, Д. Рэлея, Ф. Букингема была сформулирована в первой редакции пи - теорема, позволившая установить условия подобия явлений любой физической природы. Начиная с этого времени метод подобия, становится основным методом экстраполяции характеристик модели в характеристики оригинала при физическом моделировании.

Параллельно с развитием физического моделирования шло развитие логического моделирования в знаковой форме. История развития знакового моделирования - это, прежде всего история развитие математики. В конце XVI в. Д. Непер изобрел логарифмы, В XVII в. И. Ньютон и Г. Лейбниц создали дифференциальное исчисление. Наряду с аналитическими методами получают развитие численные методы решения различных задач. Все это привело к распространению учения о подобии на величины и процессы различной физической природы, но имеющие определенную аналогию

или хотя бы какое-то математическое соответствие. При этом стали различать подобие математическое и аналоговое. Постепенно моделирование стало охватывать все большие области научной и технической деятельности человека. Например, для отработки антисейсмичности конструкций зданий модели иногда имели довольно внушительные размеры площадью до 20 $м^2$ и массой до 30 т. Гидроэнергетические объекты, такие как плотины, каналы, гидротурбины для таких станций, как Волжская, Братская, Асуанская ГЭС, исследовались на физических моделях, изображающих в уменьшенном масштабе эти сооружения.

Широко распространены специальные модели, сочетающие в себе физическую и математическую модели с натурными приборами. Эти модели применяются для наладки приборов управления и тренировки персонала, в первом случае такие модели стали называться испытательными стендами, во втором - тренажерами.

Физическое моделирование основано на изучении явлений на моделях одной физической природы с оригиналом. При физическом моделировании сохраняют особенности поведения объекта исследования, что существенно облегчает получение требуемых результатов, так как для модели выбирают наиболее удобные геометрические размеры и диапазоны изменения физических величин.

Метод физического моделирования имеет очень важное значение, когда в комплекс явлений, характеризующих исследуемый процесс, входят такие явления, которые не поддаются математическому описанию. Одним из примеров физического моделирования является исследование переходных процессов в энергетических системах на моделях этих систем, где мощные генераторы и трансформаторы заменены малогабаритными электрическими машинами и трансформаторами, а дальние линии электропередачи - соответствующими эквивалентами. Однако во многих случаях использование метода физического моделирования приводит к необходимости изготовления дорогостоящих моделей, пригодных для решения ограниченного круга задач.

Математическое моделирование основано на идентичности дифференциальных уравнений, описывающих явление в оригинале и модели, отличающихся по своей природе. Например, математическое моделирование переходных процессов в энергетической системе может быть выполнено на электронной вычислительной машине (ЭВМ).

Главное преимущество математического моделирования перед физическим заключается в возможности исследования явлений природы, трудно поддающихся изучению, используя хорошо изученные явления. При математическом моделировании более наглядно, чем при физическом, осуществляется индикация и регистрация результатов исследований: можно просто варьировать в широких пределах исходные данные задачи для выбора оптимальных (по заданному критерию) параметров исследуемой системы, время решения задачи, по желанию исследователя, может быть изменено в широких пределах. Основу математического моделирования составляет триада модель - алгоритм - программа. Математические модели реальных исследуемых процессов сложны и включают системы нелинейных функционально-дифференциальных уравнений. Ядро математической модели составляют уравнения с частными производными.

На первом этапе вычислительного эксперимента выбирается (или строится) модель исследуемого объекта, отражающая в математической форме важнейшие его свойства - законы, которым он подчиняется, связи, присущие составляющим его частям, и т. д. Математическая модель (ее основные фрагменты) исследуется традиционными аналитическими средствами прикладной математики для получения предварительных знаний об объекте.

Второй этап связан с выбором (или разработкой) вычислительного алгоритма для реализации модели на компьютере. Необходимо получить искомые величины с заданной точностью на имеющейся вычислительной технике. Вычислительные алгоритмы должны не искажать основные свойства модели и, следовательно, исходного объекта, они должны быть адаптирующимися к особенностям решаемых задач и используемых вычислительных средств. Изучение математических моделей проводится методами вычислительной математики, основу которых составляют численные методы решения задач математической физики - краевых задач для уравнений с частными производными.

На третьем этапе создается программное обеспечение для реализации модели и алгоритма на компьютере. Программный продукт должен учитывать важнейшую специфику математического моделирования, связанную с использованием ряда (иерархии) математических моделей, многовариантностью расчетов. Это подразумевает широкое использование комплексов и пакетов

прикладных программ, разрабатываемых, в частности, на основе объектно-ориентированного программирования.

Успех математического моделирования определяется одинаково глубокой проработкой всех основных звеньев вычислительного эксперимента. Опираясь на триаду модель - алгоритм - программа, исследователь получает в руки универсальный, гибкий и недорогой инструмент, который вначале отлаживается, тестируется и калибруется на решении содержательного набора пробных задач. После этого проводится широкомасштабное исследование математической модели для получения необходимых качественных и количественных свойств и характеристик исследуемого объекта.

Вычислительный эксперимент по своей природе носит междисциплинарный характер, невозможно переоценить синтезирующую роль математического моделирования в современных научно-технических разработках. В совместных исследованиях участвуют специалисты в прикладной области, прикладной и вычислительной математике, по прикладному и системному программному обеспечению. Вычислительный эксперимент проводится с опорой на широкое использование самых разных методов и подходов - от качественного анализа нелинейных математических моделей до современных языков программирования.

Моделирование в том или ином виде присутствует почти во всех видах творческой деятельности. Математическое моделирование расширяет сферы точного знания и поле приложений рациональных методов. Оно базируется на четкой формулировке основных понятий и предположений, апостериорном анализе адекватности используемых моделей, контроле точности вычислительных алгоритмов, квалифицированной обработке и анализе данных расчетов.

Решение проблем жизнеобеспечения на современном этапе основывается на широком использовании математического моделирования и вычислительного эксперимента. Вычислительные средства (компьютеры и численные методы) традиционно хорошо представлены в естественнонаучных исследованиях, прежде всего в физике и механике. Идет активный процесс математизации химии и биологии, наук о земле, гуманитарных наук и т.д.

Наиболее впечатляющие успехи достигнуты при применении математического моделирования в инженерии и технологии.

Компьютерные исследования математических моделей в значительной степени заменили испытания моделей летательных аппаратов в аэродинамических трубах, взрывы ядерных и термоядерных устройств на полигонах.

1. МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ. ОСНОВНЫЕ ТЕРМИНЫ, ПОНЯТИЯ И ОПРЕДЕЛЕНИЯ

Модель — это физический или абстрактный образ моделируемого объекта, удобный для проведения исследований и позволяющий адекватно отображать интересующие исследователя физические свойства и характеристики объекта. Удобство проведения исследований может определяться различными факторами: легкостью и доступностью получения информации, сокращением сроков и уменьшением материальных затрат на исследование и др.

Различают моделирование предметное и абстрактное. При *предметном моделировании* строят *физическую модель,* которая соответствующим образом отображает основные физические свойства и характеристики моделируемого объекта. При этом модель может иметь иную физическую природу в сравнении с моделируемым объектом (например, электронная модель гидравлической или механической системы). Если модель и объект одной и той же физической природы, то моделирование называют *физическим.*

Физическое моделирование широко применялось до недавнего времени при создании сложных технических объектов. Обычно изготавливался макетный или опытный образец технического объекта, проводились испытания, в процессе которых определялись его выходные параметры и характеристики, оценивались надежность функционирования и степень выполнения технических требований, предъявляемых к объекту. Если вариант технической разработки оказывался неудачным, все повторялось сначала, т. е. осуществлялось повторное проектирование, изготовление опытного образца, испытания и т. д.

Физическое моделирование сложных технических систем сопряжено с большими временными и материальными затратами.

Абстрактное моделирование связано с построением *абстрактной модели.* Такая модель представляет собой математические соотношения, графы, схемы, диаграммы и т. п. Наиболее мощным и универсальным методом абстрактного моделирования является

математическое моделирование. Оно широко используется как в научных исследованиях, так и при проектировании.

Математическое моделирование позволяет посредством математических символов и зависимостей составить описание функционирования технического объекта в окружающей внешней среде, определить выходные параметры и характеристики, получить оценку показателей эффективности и качества, осуществить поиск оптимальной структуры и параметров объекта. Применение математического моделирования при проектировании в большинстве случаев позволяет отказаться от физического моделировании, значительно сократить объемы испытаний и доводочных работ, обеспечить создание технических объектов с высокими показателями эффективности и качества. С одним из основных компонентов системы проектирования. В этом случае становится математическая модель.

Математическая модель — это совокупность математических объектов и отношений между ними, адекватно отображающая физические свойства создаваемого технического объекта. В качестве математических объектов выступают числа, переменные, множества, векторы, матрицы и т. п. Процесс формирования математической модели и использования ее для анализа и синтеза называется *математическим моделированием.* В конструкторской практике под математическим моделированием обычно понимается процесс построения математической модели, а проведение исследований на модели в процессе проектирования называют *вычислительным экспериментом.* Такое деление удобно для проектировщиков и функционально вполне обосновано, поэтому в дальнейшем будем придерживаться этой терминологии.

Система – целенаправленное множество взаимосвязанных объектов любой природы, совокупность компонентов, которая рассматривается, как единое целое и организована для решения определенных функциональных задач.

Подсистемы - относительно самостоятельные части системы, функционально связанные между собой.

Элемент - компонент системы, принимаемый в данной постановке задачи как неделимый на более мелкие составляющие.

Явление - совокупность процессов, сопутствующих работе системы и проявляющихся в виде изменений состояний или режимов этой системы.

Режим - состояние системы, определяющееся множеством различных процессов и зависящее от собственных параметров системы и параметров возмущающих воздействий. Режим бывает переходным и установившимся.

Процесс - закономерное последовательное изменение относительно самостоятельной группы параметров режима, называемой параметрами процесса.

Внешняя среда – множество существующих вне объекта элементов, оказывающих влияние на исследуемый объект.

Существуют *классический* и *системный* подходы к решению задач моделирования.

Алгоритм — это предписание, определяющее последовательность выполнения операций вычислительного процесса. *Алгоритм автоматизированного проектирования* представляет собой совокупность предписаний, обеспечивающих выполнение операций и процедур проектирования, необходимых для получения проектного решения. Для наглядности алгоритмы чаще всего представляют в виде схем или графов, иногда дают их вербальное (словесное) описание. Алгоритм, записанный в форме, воспринимаемой вычислительной машиной, представляет собой *программную модель*. Процесс программирования называют *программным моделированием.*

Формализация процесса проектирования на основе математического моделирования позволяет его автоматизировать. Одним из основных компонентов *системы автоматизированного проектирования* (САПР) является *математическое обеспечение* включающее математические модели объектов проектирования и их элементов методы и алгоритмы выполнения проектных операций и процедур.

Процесс моделирования включает несколько этапов:

1 этап. Постановка задачи и определение свойств реального объекта, подлежащих исследованию.

2 этап. Констатация затруднительности или невозможности исследования реального объекта.

3 этап. Выбор модели, хорошо фиксирующей основные свойства объекта с одной стороны и легко поддающейся исследованию с другой. Модель должна отражать основные свойства объекта и не должна быть громоздкой.

4 этап. Исследование модели в соответствии с поставленной целью (проведение экспериментов).

5 этап. Проверка адекватности объекта и модели. Если нет соответствия, то необходимо повторить первые четыре этапа .

6 этап. Окончательный выбор модели.

Таким образом, моделирование состоит в выявлении основных свойств исследуемого процесса, построении моделей и их применении для прогнозирования поведения натуры. Критерием правильности моделирования является практика.

При моделировании на ЭВМ динамические характеристики, интересующие исследователя, легко и быстро воспроизводятся на устройствах отображения (осциллограф или дисплей). Этот вид моделирования можно представить как проведение определенного рода опытов средствами вычислительной техники.

Поэтому термин моделирование отражает и интерактивную форму связи человека с вычислительной машиной.

Цели моделирования:

- обоснование достоверности математических описаний;
- получение функциональных связей между величинами;
- сравнение конечного числа стратегий решения индивидуальной проблемы, т.е. ответ на вопросы: что будет, если...?;
- идентификация моделируемой системы;
- оптимизация модели. Выбор целевых функций;
- применение моделирования для обучения и тренировки.

Полностью формализовать и автоматизировать процесс проектирования практически невозможно и нецелесообразно. На этапах разработки концепции технической системы, формирования технического задания, выбора технического решения, синтеза структуры, принятия решений и др. действия конструктора, основанные на его опыте и интуиции, как правило, непредсказуемы и не поддаются формализации. САПР предусматривает тесное взаимодействие человека и ЭВМ. Это один из основополагающих принципов построения САПР. Вместе с тем все виды проектных работ, которые можно формализовать, должны быть автоматизированы. В этой связи важнейшая роль принадлежит математическому моделированию. При создании САПР необходима не только математическая модель создаваемого технического объекта, но и модели реализации всех проектных операций и процедур.

Для разработки эффективной технологии автоматизированного проектирования необходимо детальное представление обо всех этапах и стадиях создания объекта с тем, чтобы осуществить их

формализацию и математическое описание.

Наибольший эффект может дать автоматизация самых ранних этапов проектирования, когда осуществляется выбор технического решения. САПР позволяет просмотреть множество вариантов и отобрать несколько наилучших для дальнейшей Детальной проработки и окончательного выбора.

Высокий технический уровень изделия достигается в значительной мере на этапе функционального проектирования, на котором определяются основные параметры объекта. Проектные решения при этом в значительной мере определяют его качества. При недостаточной проработке проекта затраты на обеспечение качества, обусловленные необходимостью последующей доводки конструкции, достигают 10...20% от полной стоимости продукции. При этом 50...70% общих причин дефектов продукции связано с ошибками в проектно-конструкторских решениях, 20...30% с недостатками технологических процессов, 5...15% возникают по вине рабочих. Поэтому главная задача конструктора состоит в том, чтобы выявить и устранить потенциальные источники дефектов еще на стадии проектирования.

Операции и процедуры функционального проектирования, как правило, почти полностью поддаются формализации, что в конечном итоге создает необходимые условия для определения и выбора оптимальных параметров и структуры технического объекта. При этом используются математические модели создаваемых объектов, модели оценки и принятия решений, которые в виде соответствующих алгоритмов реализуются при проектировании.

При решении задач синтеза структуры, моделировании процессов функционирования объектов с переменной структурой возникает необходимость постоянного изменения математической модели. Поэтому большое внимание уделяется методам автоматизированного формирования математических моделей.

На различных этапах и стадиях проектирования сложной технической системы используются различные математические модели. На ранних стадиях обычно модели простые, но чем подробнее проработка проекта, тем сложнее нужна модель. Математические модели могут представлять собой системы дифференциальных уравнений (обыкновенных или в частных производных), системы алгебраических уравнений, простые алгебраические выражения, бинарные отношения, матрицы и др. Сложные модели требуют

больших затрат времени на проведение вычислительных экспериментов. Системы уравнений таких моделей обычно отличаются плохой обусловленностью, что создает проблемы обеспечения устойчивости вычислительного процесса, достижения необходимой точности при приемлемых затратах времени.

Поскольку все проектные работы носят оптимизационный характер, то решать системы уравнений для получения искомого результата приходится многократно. Ситуация усугубляется

также многомерностью и многокритериальностью задач. На заключительных этапах проектирования часто приходится использовать вероятностные модели, с тем, чтобы исследовать процессы функционирования технической системы в условиях, максимально приближенных к реальным.

Если САПР потребует слишком больших затрат времени на разработку проекта изделия, то она вряд ли получит широкое практическое применение.

Создание нового технического объекта — сложный и длительный процесс, в котором стадия проектирования имеет решающее значение в осуществлении замысла и достижении высокого технического уровня. Под техническим объектом в дальнейшем понимается техническая система — машина, механизм, технический комплекс, технологический процесс, а также любой их компонент, выделяемый в процессе проектирования путем декомпозиции (деления.) структуры целостного объекта на отдельные блоки, части, элементы и т. п.

Современная методология проектирования базируется на *системном подходе.* Технический объект при системном подходе рассматривается как сложная система, состоящая из взаимосвязанных, целенаправленно функционирующих элементов и находящаяся во взаимодействии с окружающей внешней средой. Это позволяет учесть все факторы, влияющие на его функционирование, и обеспечить создание технического объекта с высокими показателями эффективности и качества.

Одно из важнейших требований системного подхода заключается в необходимости рассматривать существование и функционирование технического объекта во времени и в пространстве. Описание существования объекта во времени приводит к понятию жизненного цикла, а в пространстве – к понятию внешней среды, с которой взаимодействует объект в процессе функционирования.

Жизненный цикл технического объекта представляет собой

совокупность взаимосвязанных процессов создания и последовательного изменения его состояния от формирования исходных требований к объекту до окончания его эксплуатации. Жизненный цикл состоит из следующих стадий: *создание, производство, обращение и эксплуатация.)* Каждая из стадий содержит целый ряд этапов, операций и процедур. Важно отметить, что все стадии жизненного цикла имеют прямые и обратные связи. Прямые связи очевидны. Так, качество проекта определяет надежность и эффективность технического объекта. Надежность сказывается на производственных и эксплуатационных издержках, а эффективность характеризует основные эксплуатационные свойства объекта (производительность, экономичность и др.). Но высокая эффективность новых разработок, в свою очередь, достижима лишь при учете результатов эксплуатации существующего технического объекта (или его аналога) и анализа технологических аспектов их производства. В этом случае имеют место обратные связи.

Сложность и взаимосвязанность процессов жизненного цикла требует глубокого и целенаправленного их изучения. Для этого широко используется математическое моделирование. Моделирование применяется на всех стадиях жизненного цикла. Посредством моделирования осуществляется решение исследовательских, поисковых, проектно-конструкторских и эксплуатационных задач. На этапе доводки конструкции приходится моделировать процессы функционирования технического объекта для выявления причин неудовлетворительных показателей надежности (поломки, преждевременный износ и др.) или эффективности (не достигается расчетная производительность, повышенный удельный расход энергии, низкие показатели качества переходных процессов и др.). В период эксплуатации технического объекта моделирование осуществляется с целью определения наиболее эффективных режимов функционирования, целесообразных областей и условии использования и т. п.

Процесс создания разделяется на стадии: *предпроектные исследования, техническое задание, техническое предложение, эскизный проект, технический проект, рабочий проект, изготовление опытных образцов, испытания и доводка, приемочные испытания.* Первые две стадии и частично третья составляют этап внешнего проектирования, на котором осуществляется научно-технический поиск и прогнозирование, формирование описания среды

функционирования технического объекта, моделирование и исследование, направленные на разработку концепции и технического решения. Этап внешнего проектирования, называемый также этапом научно-исследовательских работ (НИР), завершается разработкой технического задания.

Остальные стадии относятся к внутреннему проектированию и составляют этап опытно-конструкторских работ (ОКР), в процессе которого определяются и конкретизируются основные функциональные и конструктивные параметры, определяющие технико-экономические показатели и облик создаваемого технического объекта.

Решение проблемы создания нового технического объекта базируется на всесторонне обоснованной концепции и вытекает из безусловных потребностей общества, необходимости практической реализации достигнутого научного потенциала и повышения показателей эффективности.

Концепция определяется как комплекс требований к техническому объекту для выполнения его назначения и содержит описание основы функционирования объекта.

Кроме выделения стадий осуществляется *декомпозиция* процесса проектирования в зависимости от степени абстрагирования, характера отображаемых свойств объекта, его структуры, принятой схемы распределения работ между подразделениями проектно-конструкторской организации и др.

Декомпозиция приводит к выделению составных частей объекта (блоков), иерархических уровней, аспектов. Это позволяет сложную задачу проектирования свести к решению более простых задач с учетом взаимодействия между ними. Каждая задача решается на основе локальной оптимизации, но декомпозиция критериев при этом осуществляется таким образом, чтобы локальные цели были подчинены конечной цели проектирования. Следовательно, концепция системности выражается не только в выделении взаимозависимых и взаимодействующих элементов технического объекта как системы, но и в единстве целей их функционирования. Кроме того, технический объект, в свою очередь, рассматривается как элемент более сложной системы (надсистемы), в состав которой входит ряд объектов внешней среды, взаимодействующих с данным техническим объектом.

Таким образом, *методология автоматизированного проектирования базируется на системном, подходе, использующем*

принципы декомпозиции, иерархичности, итеративности, локальной оптимизации и комплексного осуществления процесса проектирования, включающего функциональный, конструкторский и технологический аспекты.

Аспекты различаются характером решаемых задач и используют различные описания.

Функциональный аспект включает отображение основных принципов функционирования, характера физических и информационных процессов в объекте. При функциональном проектировании осуществляется синтез структуры и определяются основные параметры объекта и его составных частей (элементов), оцениваются показатели эффективности и качества процессов функционирования. Результат проектирования — принципиальные, функциональные, кинематические, алгоритмические схемы и сопровождающие их документы.

Функциональное проектирование осуществляется практически на всех стадиях и этапах создания технического объекта и при этом многократно повторяется по мере раскрытия неопределенностей, характерных для начальных этапов.

Конструкторский аспект — это реализация результатов функционального проектирования. При конструкторском проектировании разрабатываются компоновки и рабочие чертежи деталей, осуществляется выбор стандартных и унифицированных элементов, материалов деталей, оформляется конструкторская и эксплуатационная документация.

При этом определяются оптимальные конструктивные параметры — размеры и форма деталей, сборочных единиц и т. п., обеспечивающие минимальные массу и габариты, равнопрочность элементов конструкции при заданном ресурсе.

Технологический аспект включает реализацию результатов конструкторского проектирования, т. е. их материализацию в виде физического изделия (машины, технической системы и т. п.). Технологическое проектирование решает задачи технологической подготовки производства. Разрабатываются технологические маршруты изготовления деталей, сборки, наладки и технологических испытаний изготавливаемых изделий, осуществляется выбор оборудования, оснастки, инструмента и т. д.

Кроме рассмотренной иерархии этапов, стадий и аспектов проектирования иерархические уровни выделяют на основе блочного

структурирования технического объекта по функциональным признакам, а также в связи с различной степенью абстрагирования при описании физических свойств технического объекта на разных этапах и стадиях проектирования.

При блочном структурировании вначале выделяют крупные блоки, составляющие верхний иерархический уровень, затем каждый блок расчленяют на более мелкие блоки, входящие в следующий уровень, и т. д. вплоть до неделимых элементов (деталей), составляющих нижний уровень иерархии. Например, блоки верхнего иерархического уровня автомобиля: двигатель, трансмиссия, ходовая часть и др. В трансмиссию входят блоки: сцепление, коробка передач, карданная передача, главная передача, дифференциал. Каждый из них может быть, в свою очередь, расчленен на более мелкие блоки.

Классический подход

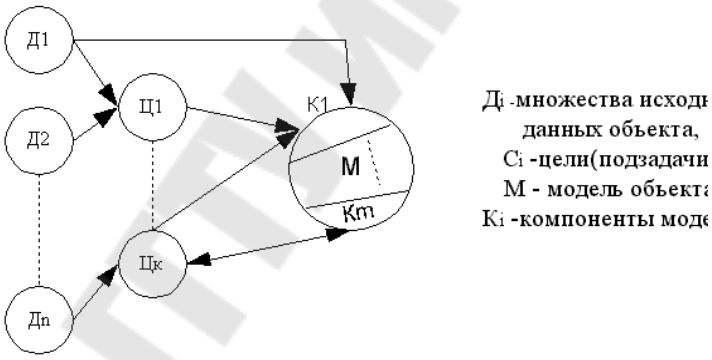


Рис.1 Схема, используемая при классическом подходе

Суть метода заключается в следующем: реальный объект, подлежащий исследованию, разбивается на отдельные компоненты $\boldsymbol{Д_i}$, и выбираются определенные цели формирования отдельных компонентов модели. Затем, на основе исходных данных, создаются компоненты модели, совокупность которых с учетом их взаимоотношений объединяется в модель.

Данный метод является индуктивным, т.е. построение модели происходит от частного к общему (от отдельных компонентов к полной модели). Классический подход используется для моделирования относительно простых систем (например, САУ).

Системный подход

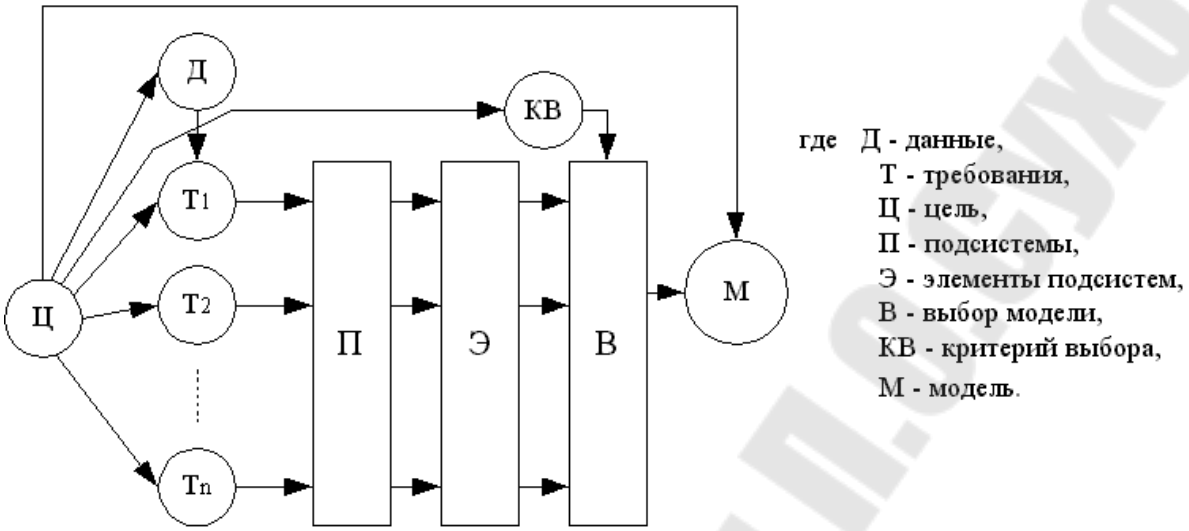


Рис. 1.2 Схема, используемая при системном подходе

Суть метода заключается в том, чтобы на основе исходных данных Д, которые известны из анализа внешней среды, с учетом ограничений, которые накладываются на систему, и в соответствии с поставленной целью формируются требования к модели объекта. На базе этих требований строятся подсистемы моделей, которые в свою очередь набираются из элементов модели. С помощью критериев выбора осуществляют выбор наилучшей модели.

Формирование модели происходит сверху, общая цель разбивается на определенные требования, по каждому требованию формируется подсистема. Системный подход очень удобен и реализуем для сложных систем.

Таблица 1

Способы создания моделей

Теоретический –предполагает создание модели на основе известных законов физики, механики, описывающих основные с точки зрения поставленной цели процессы, происходящие в объекте.	*Экспериментальный* (или идентификация) предполагает построение модели на основе результатов эксперимента, проведенного с реальным объектом.

2. ОСНОВЫ ПОСТРОЕНИЯ МАТЕМАТИЧЕСКИХ МОДЕЛЕЙ

2.1. Классификация методов моделирования

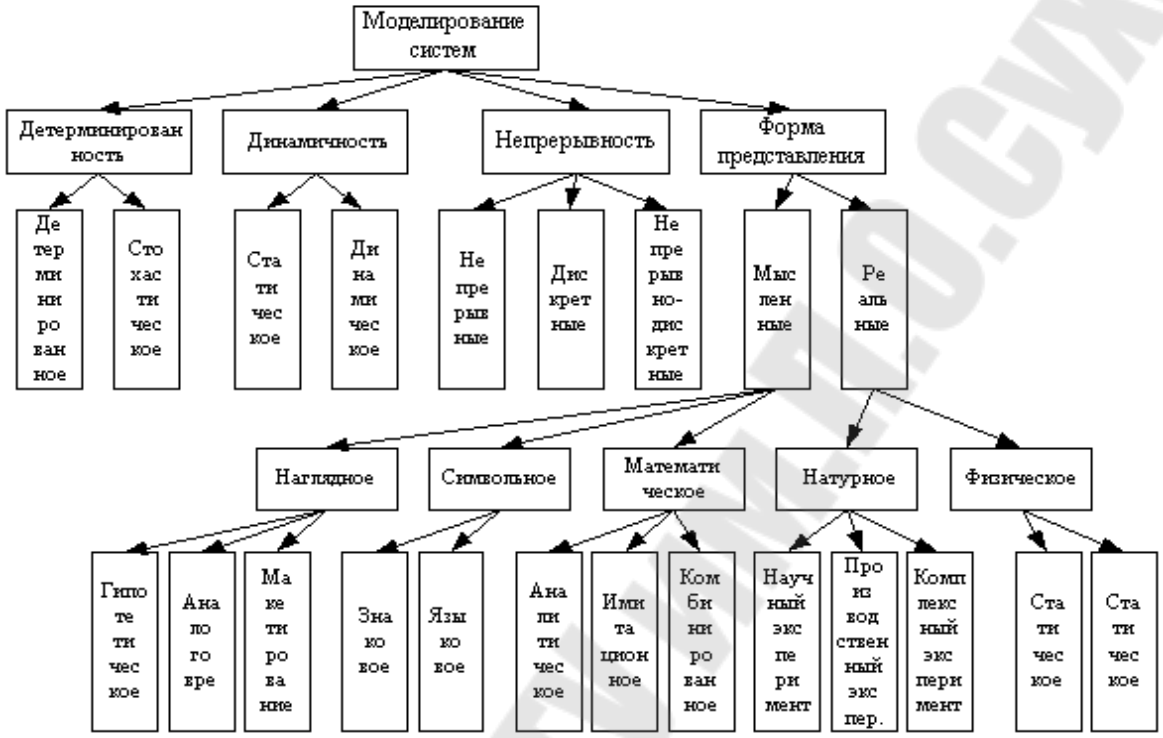


Рис. 2.1. Классификация методов моделирования

Моделирование систем включает в себя модели объекта с одной стороны и способы отражения их функционирования с другой.

По характеру изучаемых процессов моделирование может классифицироваться по следующим признакам: *детерминированность*, *динамичность*, *непрерывность* и *форма-представление*.

С точки зрения детерминированности различают: *детерминированное* и *стохастическое* моделирование. При детерминированном моделировании используются детерминированные методы без учета случайных воздействий внешней среды. *Стохастическое* моделирование отображает вероятностные и случайные процессы в объекте. При этом используется математический аппарат статистики и вероятностных процессов.

С точки зрения динамичности разделяют *статическое* и *динамичное* моделирование. *Динамичное* моделирование процессы, происходящие в объекте, рассматривает во времени. *Статическое*

моделирование изучает особые статические режимы, когда процессы, происходящие в объекте, не зависят от времени.

По признаку непрерывности различают: *непрерывное*, *дискретное* и *непрерывно-дискретное* моделирование. *Непрерывное* моделирование рассматривает процессы, происходящие в объекте, непрерывно в течение всего времени исследования. Математическим аппаратом данного типа моделирования являются дифференциальные уравнения. *Дискретное* моделирование изучает процессы в определенные моменты времени, математический аппарат – разностные уравнения. *Непрерывно-дискретное* моделирование сочетает в себе свойства непрерывного и дискретного моделирования.

По формам представления моделирование может быть *мысленное* (логическое) и *реальное* (материальное).

Мысленное моделирование применяется при исследовании систем, которые по каким-либо причинам не может быть реализовано физически. Мысленное моделирование в свою очередь разбивается на три крупных класса:

Наглядное моделирование - это создание наглядных моделей на базе представлений человека об объекте.

Наглядное моделирование подразделяется на *гипнотическое*, *аналоговое* и *макетирование*.

- *Гипнотическое моделирование* – это исследование модели в виде черного ящика, при этом структура и функциональные особенности объекта представляются гипотезой. После выдвижения гипотезы она либо принимается, либо нет.
- *Аналоговое моделирование* применяется в том случае, когда любое функциональное свойство объекта заменяется аналоговым.
- *Макетирование* применяется в случае, если невозможна физическая реализация объекта. Модель представляет собой полную аналогию с исследуемым объектом, но в другом масштабе.

Символьное моделирование – замена реального объекта неким набором символов (любому объекту ставится в соответствие символ). Выделяют *языковое* и *знаковое* моделирование.

- При *знаковом* моделировании вводятся символьные обозначения определенных понятий, однородные понятия объединяются в отдельные множества. Все знаковое моделирование сводится к теории множеств и операциям между ними.
- При *языковом* моделировании объекту и процессам, происходящим в нем, ставится в соответствие тезаурус – язык,

лишенный двусмысленности, т.е. его символика похожа на символику нашего языка, но все однозначно.

Математическое моделирование подразделяется на *аналитическое*, *имитационное* и *комбинированное*.

- *Аналитическое моделирование* – определенному объекту ставится в соответствие система уравнений и методы ее решения (высшая математика). Применяется при исследовании относительно несложных систем, к которым относится САУ.
- *Имитационное моделирование* – отдельные свойства объекта имитируются конкретными математическими способами (нет конкретной модели), используется для исследования сложных систем. Как правило, применяется к стохастическим моделям и системам массового обслуживания. Для имитационного моделирования применяется пакет GPSS.
- *Комбинированное моделирование* – это моделирование, в котором используются элементы аналитического и имитационного.

Реальное моделирование может быть *натурным* и *физическим*.

Натурное моделирование – это проведение исследований с реальными объектами с последующей обработкой результатов эксперимента.

В нем выделяют:

- *производственный эксперимент* – воспроизведение на натурном объекте основных режимов производственного процесса для дальнейшего исследования.
- *научный эксперимент* – воспроизведение на натурном объекте качественно новых режимов, увеличение технических границ.
- *комплексный эксперимент* – сочетает в себе элементы научного и производственного эксперимента

При постановке научного эксперимента реальный объект используется в качественно новых условиях функционирования или при воздействии новых факторов внешней среды с последующей обработкой результатов.

Физическое моделирование:

- в реальном масштабе времени – осуществляют постановку эксперимента в одинаковых масштабах времени как для объекта, так и для модели.
- в нереальном масштабе времени – при постановке эксперимента масштабы времени для модели и объекта различаются на некоторую величину.

2.2. Классификация математических моделей

При проектировании технических объектов используют множество видов математических моделей, в зависимости от уровня иерархии, степени декомпозиции системы, аспекта, стадии и этапа проектирования.

На любом уровне иерархии объект проектирования представляют в виде некоторой системы, состоящей из элементов. В этой связи различают *математические модели элементов* и *систем.*

При переходе к более высокому иерархическому уровню блочного структурирования система низшего уровня становится элементом системы нового уровня, и наоборот, при переходе к низшему уровню элемент становится системой. В этом случае часто оказывается нецелесообразным использование одних и тех же видов математических моделей на разных уровнях. Обычно чем ниже уровень иерархии блочного структурирования технического объекта, тем более детальное описание его физических свойств. Следовательно, на низших уровнях используют наиболее сложные математические модели. На высших уровнях могут быть с успехом применены более простые модели. Их можно получить путем аппроксимации моделей низших иерархических уровней.

В общем случае уравнения математической модели связывают физические величины, которые характеризуют состояние объекта и не относятся к перечисленным выше выходным, внутренним и внешним параметрам. Такими величинами являются: скорости и силы — в механических системах; расходы и давления — в гидравлических и пневматических системах; температуры и тепловые потоки — в тепловых системах; токи и напряжения — в электрических системах.

Величины, характеризующие состояние технического объекта в процессе его функционирования, называют *фазовыми переменными (фазовыми координатами).* Вектор фазовых переменных задает точку в пространстве, называемом *фазовым пространством.* Фазовое пространство, в отличие от геометрического, многомерное. Его размерность определяется количеством используемых фазовых координат.

Обычно в уравнениях математической модели фигурируют не все фазовые переменные, а только часть из них, достаточная для однозначной идентификации состояния объекта. Такие фазовые переменные называют *базисными координатами.* Через базисные

координаты могут быть вычислены значения и всех остальных фазовых переменных.

К математическим моделям предъявляются требования *адекватности, экономичности, универсальности.* Эти требования противоречивы, поэтому обычно для проектирования каждого объекта используют свою оригинальную модель. Модель считается адекватной, если отражает исследуемые свойства изучаемого объекта с приемлемой точностью. Точность оценивается степенью совпадения предсказанных в процессе вычислительного эксперимента на модели значений выходных параметров с истинными их значениями. Погрешность модели е по всей совокупности m учитываемых выходных параметров оценивается одной из норм вектора $\vec{\varepsilon}_M = (\varepsilon_1, \varepsilon_2, ..., \varepsilon_m)$:

$$\varepsilon = \|\vec{\varepsilon}_M\| = \max|\varepsilon_j|, j \in [1:m], \tag{2.1}$$

или

$$\varepsilon = \|\vec{\varepsilon}_M\| = \sqrt{\sum_{j=1}^{m} \varepsilon_j^2} \tag{2.2}$$

где ε_j – относительная погрешность модели по j-му выходному параметру:

$$\varepsilon_j = (\tilde{y}_j - y_j)/ y_j$$

$\tilde{y}_j$ – значение j-го выходного параметра, полученное в результате вычислительного эксперимента на принятой для проектирования математической модели; y_j – значение того же параметра, полученное при испытаниях технического объекта в контролируемых тестовых условиях или в вычислительном эксперименте на более сложной математической модели, точность которой проверена и отвечает принятой норме.

Математические модели технических объектов, используемые при проектировании, предназначены для анализа процессов функционирования объектов и оценки их выходных параметров. Они должны отображать физические свойства объектов, существенные для решения конкретных задач проектирования. При этом математическая модель должна быть как можно проще, но в то же время обеспечивать адекватное описание анализируемого процесса.

Классификация математических моделей, используемых при проектировании технических систем, приведена на рис. 3.

В зависимости от степени абстрагирования при описании физических свойств технической системы различают три основных иерархических уровня: верхний или метауровень; средний или макроуровень; нижний или микроуровень.

Метауровень соответствует начальным стадиям проектирования, на которых осуществляется научно-технический поиск и прогнозирование, разработка концепции и технического решения, разработка технического предложения. Для построения математических моделей метауровня используют методы морфологического синтеза, теории графов, математической логики, теории автоматического управления, теории массового обслуживания, теории конечных автоматов.

На *макроуровне* объект проектирования рассматривают как динамическую систему с сосредоточенными параметрами. Математические модели макроуровня представляют собой системы обыкновенных дифференциальных уравнений. Эти модели используют при определении параметров технического объекта и его функциональных элементов.

На *микроуровне* объект представляется как сплошная среда с распределенными параметрами. Для описания процессов функционирования таких объектов используют дифференциальные уравнения в частных производных. На микроуровне проектируют неделимые по функциональному признаку элементы технической системы, называемые *базовыми элементами.* Примерами таких элементов являются рамы, панели, корпусные детали, валы, диски фрикционных механизмов и др. Проектирование их основано на анализе сложно-напряженного состояния. При этом, естественно, базовый элемент рассматривается как система, состоящая из множества однотипных функциональных элементов одной и той же физической природы, взаимодействующих между собой и находящихся под воздействием внешней среды и других элементов технического объекта, также являющихся внешней средой по отношению к базовому элементу.

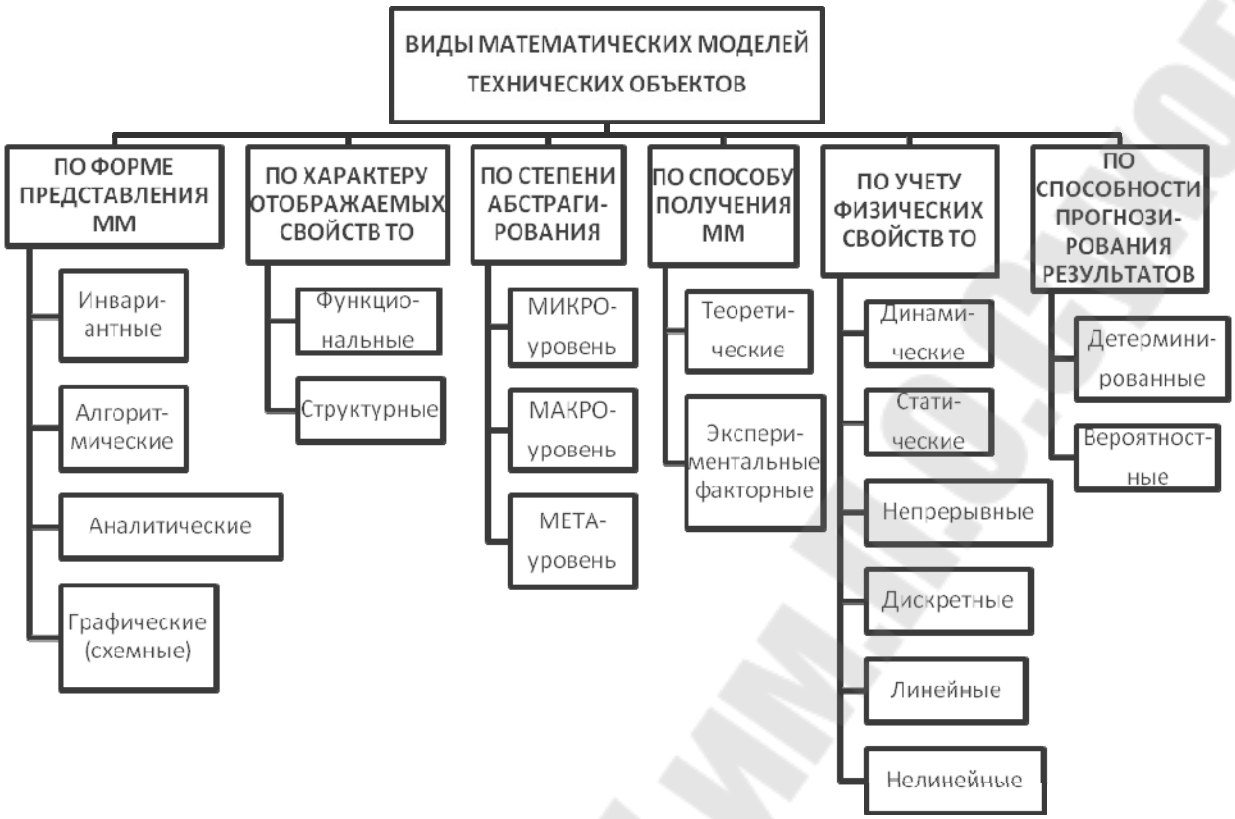


Рис. 2.2. – Классификация математических моделей.

На всех рассмотренных иерархических уровнях используют следующие виды математических моделей: детерминированные и вероятностные, теоретические и экспериментальные факторные, линейные и нелинейные, динамические и статические, непрерывные и дискретные, функциональные и структурные.

По форме представления математических моделей различают инвариантную, алгоритмическую, аналитическую и графическую модели объекта проектирования.

В *инвариантной форме* математическая модель представляется системой уравнений (дифференциальных, алгебраических), вне связи с методом решения этих уравнений.

В *алгоритмической форме* соотношения модели связаны с выбранным численным методом решения и записаны в виде алгоритма — последовательности вычислений.

Аналитическая модель представляет собой явные зависимости искомых переменных от заданных величин (обычно зависимости выходных параметров объекта от внутренних и внешних параметров). Такие модели получают на I основе физических законов, либо в результате прямого интегрирования исходных дифференциальных

уравнений, используя табличные интегралы. К ним относятся также регрессионные модели, получаемые на основе результатов эксперимента.

Графическая (схемная) модель представляется в виде графов, эквивалентных схем, динамических моделей, диаграмм и т. п. Для использования графических моделей должно существовать правило однозначного соответствия условных изображений элементов графической и компонентов инвариантной математических моделей.

Среди алгоритмических моделей выделяют *имитационные модели,* предназначенные для имитации физических и информационных процессов, протекающих в объекте при функционировании его под воздействием различных факторов внешней среды.

Математические модели могут представлять собой функциональные зависимости между выходными, внутренними и внешними параметрами:

$$\vec{Y} = \vec{F}\left(\vec{X}, \vec{Q}\right), \qquad (2.3)$$

где $\vec{Y}, \vec{X}, \vec{Q}$ - векторы выходных, внутренних, и внешних параметров, соответственно:

$\vec{Y} = \left(y_j\right),\ j = 1, m; \vec{X} = \left(x_i\right), i = 1, n; Q = \left(q_k\right),\ k = 1, l;\quad m, n, l -$ число выходных, внутренних и внешних параметров, соответственно; $\vec{F}(\bullet)$ – вектор-функция.

Математическая модель вида (3) относится к аналитической. Она позволяет легко и просто решать задачи определения оптимальных параметров. Поэтому, если представляется возможность получения модели в таком виде, ее всегда целесообразно реализовать, даже если при этом придется выполнить ряд вспомогательных процедур. Такие модели обычно получают методом планирования эксперимента (вычислительного или физического).

Деление математических моделей на функциональные и структурные определяется характером отображаемых свойств технического объекта.

Структурные модели отображают только структуру объектов и используются при решении задач структурного синтеза. Параметрами структурных моделей являются признаки функциональных или конструктивных элементов, из которых состоит технический объект и по которым один вариант структуры объекта отличается от другого. Эти параметры называют *морфологическими переменными.*

Структурные модели имеют форму таблиц, матриц и графов. Наиболее перспективно применение древовидных графов типа И-ИЛИ-дерева. Они позволяют аккумулировать накопленный опыт, используя описания всех существующих аналогов, известных из патентной литературы, и гипотетических объектов. Такие модели наиболее широко используют на метауровне при выборе технического решения.

Функциональные модели описывают процессы функционирования технических объектовов и имеют форму систем уравнений. Они учитывают структурные и функциональные свойства объекта и позволяют решать задачи как параметрического, так и структурного синтеза. Их широко используют на всех иерархических уровнях, стадиях и этапах, при функциональном, конструкторском и технологическом проектировании. На метауровне функциональные модели позволяют решать задачи прогнозирования, на макроуровне — выбора структуры и оптимизации внутренних параметров технического объекта, на микроуровне — оптимизации параметров базовых элементов и несущих конструкций.

По способам получения функциональные математические модели делятся на теоретические и экспериментальные.

Теоретические модели получают на основе описания физических процессов функционирования объекта, а *экспериментальные* — на основе изучения поведения объекта во внешней среде, рассматривая его как кибернетический "черный ящик". Эксперименты при этом могут быть *физические* (на техническом объекте или его физической модели) или *вычислительные* (на теоретической математической модели).

При построении теоретических моделей используют физический и формальный подходы.

Физический подход сводится к непосредственному применению физических законов для описания объектов, например, законов Ньютона, Гука, Кирхгофа, Фурье и др.

Формальный подход использует общие математические принципы и применяется при построении как теоретических, так и экспериментальных моделей.

Построение теоретических формальных моделей основано на вариационном принципе Гамильтона–Остроградского. Для динамических систем с сосредоточенными параметрами вариационный принцип приводит к уравнениям Лагранжа второго рода.

Экспериментальные модели — формальные. Они не учитывают всего комплекса физических свойств элементов исследуемой технической системы, а лишь устанавливают обнаруживаемую в процессе эксперимента связь между отдельными параметрами системы, которые удается варьировать и (или) осуществлять их измерение. Варьируемые параметры при этом называют факторами. Такие модели дают адекватное описание исследуемых процессов лишь в ограниченной области факторного пространства, в которой осуществлялось варьирование факторов в эксперименте. Поэтому экспериментальные математические модели носят частный характер, в то время как физические законы отражают общие закономерности явлений и процессов, протекающих как во всей технической системе, так и в каждом ее элементе в отдельности. Следовательно, экспериментальные факторные модели не могут быть приняты в качестве физических законов. Вместе с тем методы, применяемые для построения этих моделей (метод статистических испытаний, регрессионный анализ, корреляционный анализ, планирование эксперимента и др.) широко используются при проверке научных гипотез.

Функциональные математические модели могут быть линейные и нелинейные.

Линейные модели содержат только линейные функции фазовых переменных и их производных. Характеристики многих элементов реальных технических объектов нелинейные. Математические модели таких объектов включают нелинейные функции фазовых переменных и (или) их производных и относятся к *нелинейным.*

С целью упрощения задач проектирования на высших иерархических уровнях используют простые линейные модели. Если описание технического объекта представлено системой линейных обыкновенных дифференциальных уравнений, то, применяя преобразование Лапласа, ее можно привести к системе алгебраических уравнений с комплексными переменными, решение которой значительно проще, чем исходной системы дифференциальных уравнений. Такой подход используется для построения математических моделей на метауровне. В моделях макроуровня следует учитывать нелинейные свойства технического объекта.

Если при моделировании учитываются инерционные свойства технического объекта и (или) изменение во времени параметров объекта или внешней среды, то модель называют *динамической.* В

противном случае модель *статическая.* Выбор динамической или статической модели определяется режимом работы технического объекта, положенным в основу проводимой процедуры анализа в маршруте проектирования. Большинство задач функционального проектирования требует использования динамических моделей. При конструкторском проектировании часто применяют статические модели, а динамические эффекты процесса функционирования объекта учитывают при формировании нагрузочных характеристик посредством коэффициентов динамичности, определяемых в процессе функционального проектирования.

Математическое представление динамической модели в общем случае может быть выражено системой дифференциальных уравнений, а статической — системой алгебраических уравнений. Динамическая модель может также представлять собой интегральные уравнения, передаточные функции, а в аналитической форме — явные зависимости фазовых координат или выходных параметров технического объекта от времени.

Воздействия внешней среды на технический объект носят случайный характер и описываются случайными функциями.

При проектировании также учитывается случайный разброс параметров элементов объекта, обусловленный технологическим процессом изготовления. Все процессы, происходящие в объекте, также случайны и могут быть оценены вероятностными и статистическими характеристиками: вероятностью выполнения тех или иных требований, корреляционной функцией, спектральной плотностью, математическим ожиданием, дисперсией и др. Анализ функционирования объекта в этом случае требует построения *вероятностной математической модели.* Однако такая модель весьма сложная и ее использование при проектировании требует больших затрат машинного времени. Поэтому ее применяют чаще на заключительном этапе проектирования.

Большинство проектных процедур выполняется на детерминированных моделях. *Детерминированная математическая модель* характеризуется взаимно однозначным соответствием между внешним воздействием на динамическую систему и ее реакцией на это воздействие. В вычислительном эксперименте при проектировании обычно задают некоторые стандартные типовые воздействия на объект: *ступенчатыми, импульсными, гармоническими, кусочно-линейными, экспоненциальными* и др. Их называют *тестовыми воздействиями.*

3 ЧИСЛЕННЫЕ МЕТОДЫ РЕШЕНИЯ ИНЖЕНЕРНЫХ ЗАДАЧ

Численные методы применяются для решения задач, которые не удается решить методами классической математики, оперируют числами и используют простейшие арифметические действия: сложение, вычитание, умножение, деление, логарифмирование, т.е. именно те действия, которые реализуют процессоры большинства современных ЭВМ. Результатом численного решения задачи является число или совокупность чисел, причем это решение, как правило, не точное, приближенное.

Подавляющее большинство численных методов предусматривает замену исходной задачи другой задачей, математически более простой, решение которой можно получить с помощью арифметических действий. Поэтому главной проблемой при использовании численных методов является доказательство того факта, что численное решение совпадает с решением исходной задачи, либо разница между ними не превышает заранее указанного значения.

3.1 Погрешности решения задач с помощью ЭВМ

Численное решение задачи с помощью ЭВМ практически никогда не бывает абсолютно точным. Оценке погрешности результатов решения задач придается очень большое значение, так как приближенное решение ценно лишь тогда, когда известна его точность, т.е. известно, до какой степени ему можно доверять. Основные источники появления погрешностей:

1 *Использование численных методов.* Точных численных методов, у которых отсутствует погрешность, практически нет. Чаще всего при численном решении задач заранее задается допускаемая погрешность с учетом требуемой точности решения и имеющихся ресурсов (мощности используемой ЭВМ и отведенного времени).

2 *Конечная аппроксимация бесконечных процессов*. Вручную легко взять производную от элементарной функции, т.е. получить ее аналитическое выражение. На ЭВМ можно лишь приближенно определить *значение* производной функции в заданной точке. По определению

$$f'(x_0) = \lim_{\Delta x \to 0} \frac{[f(x_0 + \Delta x) - f(x_0)]}{\Delta x},$$

но на ЭВМ нельзя получить бесконечно малое число, поэтому используется конечное число Δx, величина которого зависит от требуемой точности вычисления производной и возможностей разрядной сетки ЭВМ.

3 *Ошибки округления при расчетах на ЭВМ и их накопление*. Разрядная сетка любой ЭВМ конечна, поэтому периодические дроби и числа с большим количеством знаков после запятой автоматически округляются (уменьшаются). Любое арифметическое действие с такими числами увеличивает ошибку округления.

4 *Ошибки в исходных данных.* В число исходных данных для решения практических задач, как правило, входят результаты экспериментов, точность которых определяется выбранной методикой их получения и обработки, а также качеством приборов.

Погрешности видов 2-4 неустранимы. Их исследование может помочь в выборе метода решения задачи: нецелесообразно использовать метод, погрешность которого существенно меньше неустранимой. Погрешность аппроксимации бесконечных процессов не должна превышать погрешность метода. Для уменьшения накопления ошибок округления следует по возможности избегать вычитания близких чисел, т.к. относительная погрешность результата $\delta_{(a_1-a_2)}=\frac{\Delta a_1+\Delta a_2}{|a_1-a_2|}$ при этом существенно возрастает, и сводить число арифметических действий при решении задачи к минимуму.

Причиной погрешностей численного решения задачи также может быть ее некорректная постановка. Задача поставлена корректно, если ее решение существует, единственно и устойчиво. Существование решения гарантировано, если при переходе от словесной постановки задачи к математическому описанию не потерян ее физический смысл. Единственность определяется правильностью задания области определения задачи. Устойчивость означает, что при малых изменениях исходных данных и результат меняется незначительно. Проблема устойчивости существует и для алгоритмов. Неудачный алгоритм, даже при правильно выбранном методе, может дать неверный результат.

3.2 Приближенное решение нелинейных уравнений

В нормальном виде нелинейное уравнение записывается следующим образом: $f(x)=0$. Если функция $f(x)$ имеет вид

$f(x) = a_0 + a_1 x + a_2 x^2 + ... + a_n x^n$ $f(x)=a_0+a_1x+a_2x^2+...+a_nx^n$, где a_0, a_1, …, a_n - любые действительные числа, то уравнение называется алгебраическим. Уравнение называется трансцендентным, если $f(x)$ включает функции a^x, $\log_a x$, $\sin(x)$, $\mathrm{tg}(x)$ и т.п.

Постановка задачи: Найти такое действительное число (числа) x^*, что $f(x^*) \equiv 0$. В инженерной практике такие задачи возникают при выполнении технологических и механических расчетов машин и аппаратов, расчетов систем автоматического регулирования, собственных колебаний конструкций со многими степенями свободы, равновесных концентраций гомогенных химических реакций. Коэффициенты реальных уравнений, как правило, определяются приближенно (на основе данных экспериментов), поэтому на практике точки x^* также определяются приближенно: осуществляется поиск точки, отстоящей от x^* менее чем на заданную величину ε.

Процесс численного решения уравнений вида $f(x)=0$ включает два этапа: 1) отделение искомого корня, т.е. определение отрезка $[a;b]$ оси x, содержащего этот корень; 2) уточнение корня до заданной степени точности ε, т.е. поиск точки x, такой, что $|x - x^*| < \varepsilon$.

3.2.1 Отделение корней нелинейных уравнений

Теоретической основой отделения корней уравнения $f(x) = 0$ является *II теорема Вейерштрасса*: внутри отрезка $[a;b]$ оси x имеется единственный корень уравнения $f(x) = 0$, если:

1) функция $f(x)$ непрерывна внутри отрезка $[a;b]$;

2) на концах отрезка $f(x)$ имеет значения разных знаков: $f(a)\cdot f(b) < 0$;

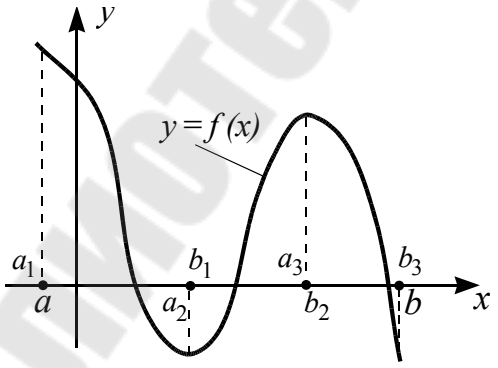


Рис. 3.1 Иллюстрация к II теореме Вейерштрасса

3) $f'(x)$ знакопостоянна на отрезке $[a;b]$: $\forall x \in [a;b] \quad f'(x) > 0$ или $f'(x) < 0$.

Если непрерывная при $x \in [a;b]$ функция $f(x)$ на концах отрезка имеет значения разных знаков, то ее график, по крайней мере, один раз пересекает ось x (см. рис. 3.1). Точка пересечения будет единственной, если $f(x)$ на всем отрезке $[a;b]$ только

возрастает ($f'(x)>0$) или только убывает ($f'(x)<0$).

На практике обычно используют графический способ отделения корней:

1) строится график функции $y = f(x)$ (например, средствами MathCAD);

2) выделяются отрезки оси x, включающие абсциссы точек пересечения графика с осью;

3) для найденных отрезков проверяется выполнение условий II теоремы Вейерштрасса.

Пример. Отделить максимальный корень уравнения $x^2 + x - 2 = 0$.

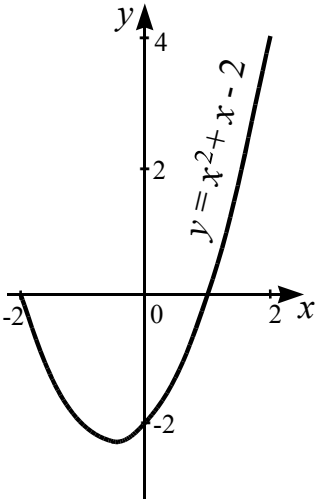


Рис. 3.2 График функции $y = x^2 + x - 2$

Согласно графику (рис. 3.2), максимальный корень уравнения лежит в отрезке [0;2]. Проверим условия теоремы Вейерштрасса для этого отрезка:

1) $f(x)$ непрерывна при $x \in [0;2]$;

2) $f(0)=-2 < 0$, $f(2)=4 > 0 \rightarrow f(0) \cdot f(2) < 0$;

3) $f'(x)=2x + 1$, $f'(x) > 0 \ \forall x \in [0;2]$.

Условия выполняются, поэтому $x^* \in [0;2]$.

Один из наиболее популярных алгоритмов отделения корня сводится к следующему: аргумент x изменяется с малым шагом от левой границы выбранного отрезка оси x к правой и на каждом шаге проверяется изменение знака $f(x)$ и постоянство знака $f'(x)$. Если производная меняет знак, а функция – нет, начальная точка переносится в текущую. При реализации этого алгоритма рекомендуется учитывать условия применимости метода уточнения корня, который предполагается использовать.

3.2.2 Уточнение корней нелинейных уравнений

К числу наиболее популярных методов уточнения корня уравнения

$f(x) = 0$ внутри отрезка $[a;b]$ оси x относятся метод бисекции, метод Ньютона и метод простых итераций.

Метод бисекции (деления отрезка пополам). Для уточнения корня используется условие 2) теоремы Вейерштрасса:

а) отрезок $[a;b]$ оси x, содержащий искомый корень уравнения, делится пополам (рис. 3.3) и вычисляется значение $f(x)$ в точке

$c=(a+b)/2$; если $f(c)=0$, то корень найден: $x^*=c$;

б) при $f(c)\neq 0$ и $f(a)\cdot f(c)<0$ от отрезка $[a;b]$ отсекается правая половина, т.е. $b\to c$, а если $f(b)\cdot f(c)<0$, то левая, т.е. $a\to c$;

в) если $|b-a|<\varepsilon$ (заданная точность), то для нового отрезка $[a;b]$ вновь выполняются п.п. а),б), в противном случае процесс уточнения корня прекращается и принимается: $x^*\approx (a+b)/2$ (середина последнего отрезка).

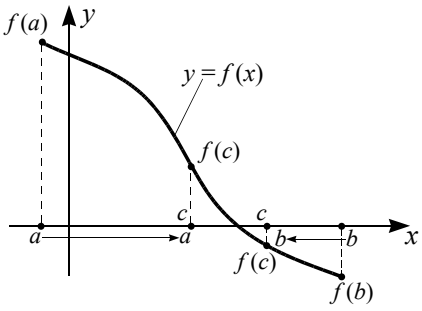


Рис.3.3 Иллюстрация к методу бисекции

Алгоритм метода бисекции сходится для любой функции, непрерывной на отрезке $[a;b]$ оси x, за $N=\log_2[(b-a)/\varepsilon]$ шагов. Он абсолютно надежен, но обладает малой скоростью сходимости.

Пример. Отделить корень уравнения $\sqrt{x-1}-1.5=0$ и уточнить его методом бисекции.

Отделение корня: Область определения $f(x)$: $x\geq 1$; $f'(x)=1/(2\sqrt{x-1})$: при $x>1$ $f'(x)>0$ или <0 в зависимости от знака числа, извлекаемого из-под корня, т.е. уравнение может иметь один корень. $f(1)=-1.5<0$, $f(5)=\pm 2-1.5$, т.е., если принять $\sqrt{4}=+2$, то $f(1)\cdot f(5)<0\to x^*\in[1;5]$.

Уточнение корня: 1) $c=(1+5)/2=3$; $f(3)=\sqrt{2}-1.5\approx -0{,}1\to x^*\in[3;5]$;

2) $c=(3+5)/2=4$; $f(4)=\sqrt{3}-1.5\approx 0.2$, $\to x^*\in[3;4]$; 3) $c=(3+4)/2=3.5$; $f(3.5)=\sqrt{2.5}-1.5\approx 0.1\to x^*\in[3;3.5]$; $\varepsilon_{n=3}=|3.5-3|=0.5$, $x^*\approx(3+3.5)/2=3.25$.

Метод Ньютона (касательных). Порядок уточнения корня с помощью этого метода иллюстрирует рис. 3.4:

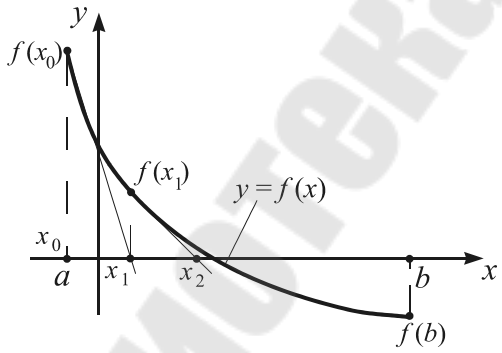


Рис. 3.4 Иллюстрация к методу Ньютона

1) внутри отрезка $[a;b]$ оси x, содержащего искомый корень уравнения $f(x)=0$, выбирается точка x_0 - начальное приближение;

2) в точке $(x_0, f(x_0))$ проводится касательная к графику функции $f(x)$: $y-f(x_0)=f'(x_0)(x-x_0)$, - и определяется точка ее пересечения с осью x: в этой точке $y=0$, поэтому $x_1=x_0-f(x_0)/f'(x_0)$;

3) процесс построения касательных и определения точек их пересечения с осью x по правилу $x_{k+1}=x_k-f(x_k)/f'(x_k),\ k=0,1,2,...$ продолжается до тех пор, пока не выполнится условие:

$(x_{k+1}-x_k)^2 < \varepsilon\cdot(2m_1/M_2)$, где $m_1=\min\limits_{x\in[a;b]}|f'(x)|$, $M_2=\max\limits_{x\in[a;b]}|f''(x)|$.

Условие применимости метода: Функция $f(x)$ не должна иметь внутри отрезка $[a;b]$ оси x точек перегиба (либо выпукла, либо вогнута), то есть $f''(x)$ должна быть знакопостоянна $\forall x\in[a;b]$. В этом случае все касательные лежат по одну сторону графика функции $f(x)$, следовательно знаки $f(x_k)$, $k=0,1,2,\ldots$ одинаковы и нет необходимости проверять смену знака $f(x)$ на $[a;x_k]$ и $[x_k;b]$.

С учетом условия применимости метода рекомендуется выбирать начальное приближение следующим образом: $x_0=a$, если $f(a)\cdot f''(a)>0$ и $x_0=b$, если $f(b)\cdot f''(b)>0$ (при $f(x_0)\cdot f''(x_0)<0$ касательная может пересечь ось x за пределами отрезка $[a;b]$, где не проверялись условия отделения).

По скорости алгоритм метода Ньютона существенно превосходит алгоритм метода бисекции, но область его применения ограничена функциями, не имеющими внутри отрезка $[a;b]$ точек перегиба.

Метод простых итераций (последовательных приближений) предусматривает: 1) преобразование исходного уравнения $f(x)=0$ к эквивалентному виду $x=\varphi(x)$; 2) выбор начального приближения $x_0\in[a;b]$ (обычно $x_0=a$ или $x_0=b$); 3) организацию итерационного процесса: $x_1=\varphi(x_0),\quad x_2=\varphi(x_1),\ldots,\quad x_n=\varphi(x_{(n-1)}),\ldots,$ сходящегося при $n\to\infty$ к решению исходного уравнения.

В зависимости от вида функции $\varphi(x)$ итерационный процесс может как сходиться к т. x^* так и расходиться (см. рис. 3.5). Если процесс сходится, то: $|x_{k-1}-x_{k-2}|<|x_{k-1}-x_k|$, $k=1,2,\ldots$ Для вывода условия сходимости воспользуемся разложением функции $\varphi(x)$ в ряд Тейлора в окрестности точки x_k:

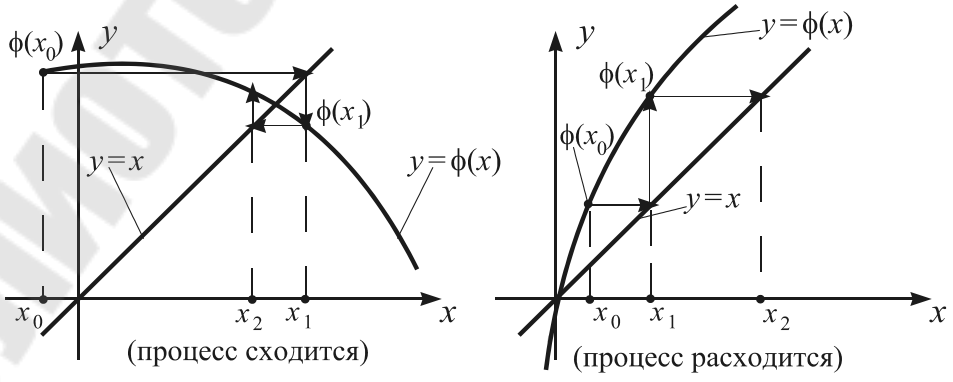


Рис 3.5. Иллюстрации к методу простых итераций

Таким образом, процесс $x_k = \varphi(x_{k-1})$, $k = 1,2,...$ сходится к решению x^* исходного уравнения, если: а) функция $\varphi(x)$ определена и непрерывно-дифференцируема в некоторой окрестности т. x^*; б) все приближения x_k, $k = 0,1,2,...$ лежат в этой окрестности; в) $|\varphi'(x_k)| < 1$, $k = 0,1,2,....$ Это условие достаточное, но не необходимое, т.е. при его выполнении сходимость наблюдается всегда, но может наблюдаться и при невыполнении.

Скорость сходимости итераций тем больше, чем меньше $|\varphi'(x_k)|$. Условие прекращения итерационного процесса: $|x_{k+1} - x_k| < \varepsilon \cdot \frac{1-q}{q}$, где ε - заданная точность, $q = \max\limits_{x \in [a;b]} |\varphi'(x)|$. При этом $x^* \approx x_{k+1}$. На практике чаще используют упрощенное условие: $|x_{k+1} - x_k| < \varepsilon$.

Достоинство метода - простота. Недостатки: неочевидный выбор функции $\varphi(x)$ и малая скорость сходимости при $|\varphi'(x)| \approx 1$.

3.3 Методы численного решения систем уравнений

Технологические и механические расчеты машин и аппаратов, расчеты систем автоматического регулирования, собственных колебаний конструкций, равновесных концентраций гомогенных химических реакций достаточно часто требуют решения нескольких нелинейных уравнений. Системы уравнений вида $\frac{\partial f(x_1,x_2,...,x_n)}{\partial x_i} = 0$, $i = 1,2,...,n$ приходится решать при поиске экстремумов функций $f(x_1,x_2,...,x_n)$. К решению систем линейных уравнений сводятся задачи приближения экспериментальных зависимостей аналитическими функциями, решения систем нелинейных уравнений, дифференциальных уравнений в частных производных.

Постановка задачи: Найти такие действительные числа $x_1^*, x_2^*, ..., x_n^*$, что все уравнения системы

$$\begin{cases} f_1(x_1, x_2, ..., x_n) = 0; \\ f_2(x_1, x_2, ..., x_n) = 0; \\ ..... \\ f_n(x_1, x_2, ..., x_n) = 0; \end{cases} \qquad (3.1)$$

одновременно обращаются в тождества. Другая форма записи системы (3.1):

$$f_i(x_1,x_2,...,x_n) = 0, \quad i = 1,2,...,n. \qquad (3.2)$$

Соотношения (3.1) - (3.2) - это запись системы уравнений в

нормальном виде.

Наиболее популярные численные методы решения систем уравнений: *метод простых итераций*, *метод Зейделя* (используются для решения как систем линейных, так и нелинейных уравнений), различные модификации *метода Гаусса* (для решения систем линейных уравнений), *метод Ньютона* (для решения систем нелинейных уравнений).

3.3.1 Порядок применения методов простых итераций

Исходная система преобразуется к эквивалентному виду: $x_i = \varphi_i(x_1, x_2, ..., x_n)$, i = 1,2,...,n (из одного уравнения выражается x_1, из другого – x_2 и т.д.). Выбираются начальные значения неизвестных: $x_i^{(0)}$, i=1,2,...,n и реализуется итерационный процесс вычисления приближений к решению системы по правилам:

метод простых итераций - $x_i^{(k+1)} = \varphi_i(x_1^{(k)}, x_2^{(k)}, ..., x_n^{(k)})$, k=0,1,2,...; i=1,2,...,n;

метод Зейделя - $x_i^{(k+1)} = \varphi_i(x_1^{(k+1)}, ..., x_{i-1}^{(k+1)}, x_i^{(k)}, ..., x_n^{(k)})$, k=0,1,2,...; i=1,2,...,n.

Условие прекращения процесса:

для систем линейных уравнений – $$\max_{i=1,...n}\left\{\left|\frac{x_i^{(k+1)} - x_i^{(k)}}{x_i^{(k+1)}}\right|\right\} < \varepsilon ; \qquad (3.3)$$

для систем нелинейных уравнений – $$\max_{i=1,...,n}\left\{\left|x_i^{(k+1)} - x_i^{(k)}\right|\right\} < \varepsilon \qquad (3.4)$$

(ε - заданная точность решения), при этом $x_i^* \approx x_i^{(k+1)}$, i = 1,2,...,n.

Отличие метода Зейделя от метода простых итераций: при вычислении $x_2^{(k+1)}$ вместо $x_1^{(k)}$ используется только что вычисленное значение $x_1^{(k+1)}$; $x_3^{(k+1)}$ вычисляется уже с использованием вычисленных в текущей итерации значений $x_1^{(k+1)}$, $x_2^{(k+1)}$ и т.д. Такой прием позволяет увеличить скорость сходимости итераций, т.е. обеспечить выполнение условий (1.3)-(1.4) при меньшем значении k. Преимущество метода Зейделя в скорости тем больше, чем больше порядок рассматриваемой системы.

Достаточное условие сходимости итерационных процессов:

1) функции $\varphi_i(x_1, x_2, ..., x_n)$, i=1,2,...,n определены и непрерывно дифференцируемы в некоторой окрестности решения системы;

2) начальное $x_i^{(0)}, i=1,2,...n$ и все вычисляемые приближения к решению системы находятся в этой окрестности;

3) во всех точках окрестности одновременно выполняются неравенства

$$\sum_{j=1}^{n}\left|\frac{\partial\varphi_i(x_1,x_1,...,x_n)}{\partial x_j}\right|<1,\ i=1,2,...,n. \qquad (3.5)$$

В вычислительной практике методы простых итераций и Зейделя применяются для решения систем нелинейных уравнений сравнительно редко – когда порядок системы не превышает 3. С увеличением порядка становится все сложнее проверять условие сходимости итераций, т.е. формировать и решать систему нелинейных неравенств (3.5). Гораздо чаще эти методы применяют для решения систем линейных уравнений:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + ... + a_{1n}x_n = b_1; \\ a_{21}x_1 + a_{22}x_2 + ... + a_{2n}x_n = b_2; \\ ..... \\ a_{n1}x_1 + a_{n2}x_2 + ... + a_{nn}x_n = b_n; \end{cases}, \qquad (3.6)$$

где a_{ij}, b_i, $i, j = 1,2,..., n$ - действительные числа.

Наиболее простой способ преобразования уравнений линейных систем к эквивалентному виду: $x_i = \varphi_i(x_1, x_2,..., x_n) = \frac{1}{a_{ii}}(b_i - \sum_{j=1, j\neq i}^{n} a_{ij}x_j), \quad i = 1,2,..., n$, тогда итерационный процесс решения системы примет вид:

для метода итераций (Якоби) –

$$x_i^{(k+1)} = \frac{1}{a_{ii}}(b_i - \sum_{j=1, j\neq i}^{n} a_{ij}x_j^{(k)}), \quad k = 0,1,...; \quad i = 1,2,..., n;$$

для метода Зейделя –

$$x_i^{(k+1)} = \frac{1}{a_{ii}}(b_i - \sum_{j=1}^{i-1} a_{ij}\cdot x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij}x_j^{(k)}), \quad k = 0,1,...; \quad i = 1,2,..., n.$$

Поскольку $\frac{\partial \varphi_i(x_1, x_2,..., x_n)}{\partial x_j} = \frac{a_{ij}}{a_{ii}}, \ i, j = 1,2,..., n; \ j \neq i$, условие (3.5) сходимости итерационного процесса примет вид $\sum_{j=1, j\neq i}^{n} |a_{ij}/a_{ii}| < 1$ или

$$|a_{ii}| > \sum_{j=1, j\neq i}^{n} |a_{ij}|, \ i=1,2,...,n, \qquad (3.7)$$

т.е. диагональные коэффициенты матрицы системы по абсолютной величине должны превосходить сумму абсолютных величин всех других коэффициентов соответствующего уравнения.

При решении линейных систем обычно принимают $x_i^{(0)} = 0$, $i =$

1,…,*n*.

3.3.2 Метод Гаусса

В отличие от методов Якоби и Зейделя, которые относятся к группе итерационных, метод Гаусса - прямой метод решения систем линейных уравнений. С его помощью можно получить абсолютно точное решение системы (без учета ошибок округления, свойственных ЭВМ). В основе метода Гаусса лежит идея последовательного исключения неизвестных из уравнений системы.

Наиболее распространенная схема решения систем вида (3.6) методом Гаусса предусматривает два этапа: *прямой ход*, в результате которого матрица системы преобразуется в треугольную, и *обратный ход* - вычисление неизвестных, начиная с x_n до x_1. Прямой ход начинается с вычисления для системы (3.6) множителей $m_i^{(1)} = -a_{i1}/a_{11}$, $i=2, 3,\ldots, n$ и сложения каждого i-го уравнения с первым, умноженным на $m_i^{(1)}$. Система примет вид

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \ldots + a_{1n}x_n = b_1; \\ 0\cdot x_1 + a_{22}^{(1)}x_2 + a_{23}^{(1)}x_3 + \ldots + a_{2n}^{(1)}x_n = b_2^{(1)}; \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ 0\cdot x_1 + a_{n2}^{(1)}x_2 + a_{n3}^{(1)}x_3 + \ldots + a_{nn}^{(1)}x_n = b_n^{(1)}; \end{cases},$$

где $a_{ij}^{(1)} = a_{ij} + m_i^{(1)}\cdot a_{1j}$; $b_i^{(1)} = b_i + m_i^{(1)}\cdot b_1$; $i = 2, 3,\ldots, n$; $j = 1, 2,\ldots, n$; т.е. неизвестное x_1, будет исключено из всех уравнений, кроме первого. Аналогично x_2 исключается из всех уравнений, кроме 1-го и 2-го, x_3 - из всех, кроме первых трех и т.д. При исключении x_k формируются множители

$m_i^{(k)} = -a_{ik}^{(k-1)}/a_{kk}^{(k-1)}$; $i = k+1, k+2,\ldots, n$ и производится пересчет значений $a_{ij}^{(k)} = a_{ij}^{(k-1)} + m_i^{(k)}\cdot a_{kj}^{(k-1)}$; $b_i^{(k)} = b_i^{(k-1)} + m_i^{(k)}\cdot b_k^{(k-1)}$; $i = k+1, k+2,\ldots, n$;

$j = k, \ldots, n$. После исключения x_{n-1} из последнего уравнения система примет вид

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \ldots + a_{1n}x_n = b_1; \\ \qquad a_{22}^{(1)}x_2 + a_{23}^{(1)}x_3 + \ldots + a_{2n}^{(1)}x_n = b_2^{(1)}; \\ \qquad\qquad a_{33}^{(2)}x_3 + \ldots + a_{3n}^{(2)}x_n = b_3^{(2)}; \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ \qquad\qquad\qquad\qquad a_{nn}^{(n-1)}x_n = b_n^{(n-1)}; \end{cases},$$

т.е. ее матрица станет треугольной.

Обратный ход:

$$x_n = b_n^{(n-1)} / a_{nn}^{(n-1)}, \;\; x_{n-1} = (b_{n-1}^{(n-2)} - a_{n,n-1}^{(n-2)} \cdot x_n) / a_{n-1,n-1}^{(n-2)},$$

$$x_j = (b_j^{(j-1)} - a_{jn}^{(j-1)} \cdot x_n - ... - a_{j,j+1}^{(j-1)} \cdot x_{j+1}) / a_{jj}^{(j-1)}, \;\; j = n-2, n-3, ...1.$$

Довольно популярен алгоритм решения систем линейных уравнений, основанный на *схеме метода Гаусса без обратного хода*: при исключении неизвестных матрица системы преобразуется не к треугольному, а к диагональному виду. Для этого выполняются следующие действия:

- цикл по k=1, 2,…, n;
- цикл по i=1, 2,…, k-1, k+1,…, n;
- внутри цикла по i: $m=-a_{ik}/a_{kk}$, $a_{ik}=0$, $b_i=b_i+m\cdot b_k$;
- цикл по j=1, 2,…, k-1, k+1,…, n;
- внутри цикла по j: $a_{ij}=a_{ij}+m\cdot a_{kj}$;
- окончание циклов по j, i, k.

Вычисление неизвестных: $x_i=b_i/a_{ii}$ в цикле по i=1, 2,…, n.

Метод Гаусса можно применять при условии, что все диагональные коэффициенты матрицы системы ненулевые.

Если в процессе исключения неизвестных какой-либо диагональный коэффициент обратится в 0, значит одно из уравнений системы является линейной комбинацией других и рассматриваемая система вырождена.

Область применения Метода Гаусса шире, чем итерационных методов: с его помощью можно решать системы, для которых не выполняется достаточное условие сходимости итераций (3.7).

При решении систем линейных уравнений методом Гаусса на ЭВМ для уменьшения ошибки округления уравнения следует переставить таким образом, чтобы диагональные коэффициенты матрицы системы были наибольшими по абсолютной величине в соответствующих столбцах.

3.3.3 Метод Ньютона

Метод Ньютона - наиболее популярный численный метод решения систем нелинейных уравнений. Он предусматривает выделение из уравнений системы линейных частей, которые играют определяющую роль при малых приращениях аргументов. В

результате решение системы нелинейных уравнений сводится к решению последовательности систем линейных уравнений.

Рассмотрим схему метода Ньютона на примере системы двух уравнений, приведенной к нормальному виду: $\begin{cases} f_1(x_1,x_2)=0; \\ f_2(x_1,x_2)=0; \end{cases}$, и предположим, что функции $f_1(x_1,x_2)$ и $f_2(x_1,x_2)$ непрерывно-дифференцируемы в заданной области изменения значений x_1, x_2 и начальные значения неизвестных $x_1^{(0)}, x_2^{(0)}$ принадлежат этой области.

Выделение линейных частей уравнений системы осуществляется путем разложения функций $f_1(x_1,x_2)$, $f_2(x_1,x_2)$ в ряд Тейлора в окрестности имеющегося приближения к решению системы $(x_1^{(k)}, x_2^{(k)})$, $k = 0,1,\ldots$ и отбрасывания слагаемых второго и более высоких порядков:

$$\begin{cases} \dfrac{\partial f_1(x_1^{(k)}, x_2^{(k)})}{\partial x_1} \cdot (x_1 - x_1^{(k)}) + \dfrac{\partial f_1(x_1^{(k)}, x_2^{(k)})}{\partial x_2} \cdot (x_2 - x_2^{(k)}) = -f_1(x_1^{(k)}, x_2^{(k)}); \\ \dfrac{\partial f_2(x_1^{(k)}, x_2^{(k)})}{\partial x_1} \cdot (x_1 - x_1^{(k)}) + \dfrac{\partial f_2(x_1^{(k)}, x_2^{(k)})}{\partial x_2} \cdot (x_2 - x_2^{(k)}) = -f_2(x_1^{(k)}, x_2^{(k)}). \end{cases}$$

Решением этой системы *линейных уравнений* (например, по правилу Крамера), будет следующее приближение к решению исходной системы $(x_1^{(k+1)}, x_2^{(k+1)})$: $x_1^{(k+1)} = x_1^{(k)} + \Delta_1^{(k)} / \Delta^{(k)}$, $x_2^{(k+1)} = x_2^{(k)} + \Delta_2^{(k)} / \Delta^{(k)}$, где

$$\Delta^{(k)} = \begin{vmatrix} \dfrac{\partial f_1(x_1^{(k)}, x_2^{(k)})}{\partial x_1} & \dfrac{\partial f_1(x_1^{(k)}, x_2^{(k)})}{\partial x_2} \\ \dfrac{\partial f_2(x_1^{(k)}, x_2^{(k)})}{\partial x_1} & \dfrac{\partial f_2(x_1^{(k)}, x_2^{(k)})}{\partial x_2} \end{vmatrix}; \quad \Delta_1^{(k)} = \begin{vmatrix} -f_1(x_1^{(k)}, x_2^{(k)}) & \dfrac{\partial f_1(x_1^{(k)}, x_2^{(k)})}{\partial x_2} \\ -f_2(x_1^{(k)}, x_2^{(k)}) & \dfrac{\partial f_2(x_1^{(k)}, x_2^{(k)})}{\partial x_2} \end{vmatrix};$$

$$\Delta_2^{(k)} = \begin{vmatrix} \dfrac{\partial f_1(x_1^{(k)}, x_2^{(k)})}{\partial x_1} & -f_1(x_1^{(k)}, x_2^{(k)}) \\ \dfrac{\partial f_2(x_1^{(k)}, x_2^{(k)})}{\partial x_1} & -f_2(x_1^{(k)}, x_2^{(k)}) \end{vmatrix}.$$

Новые приближения вычисляются до тех пор, пока не выполнится условие: $\max\{(x_1^{(k+1)} - x_1^{(k)})^2, (x_2^{(k+1)} - x_2^{(k)})^2\} < \varepsilon$. Решение исходной системы с точностью ε будет найдено, если все последовательно формируемые системы линейных уравнений будут невырождены, т.е. $\Delta^{(k)} \neq 0$, $k = 0,1,\ldots$

При использовании метода Ньютона для решения системы (3.1) придется последовательно формировать и решать системы линейных уравнений:

$$\begin{cases} \dfrac{\partial f_1(x_1^{(k)},...,x_n^{(k)})}{\partial x_1}\cdot(x_1-x_1^{(k)})+...+\dfrac{\partial f_1(x_1^{(k)},...,x_n^{(k)})}{\partial x_n}\cdot(x_n-x_n^{(k)})=-f_1(x_1^{(k)},...,x_n^{(k)}) \\ \dfrac{\partial f_2(x_1^{(k)},...,x_n^{(k)})}{\partial x_1}\cdot(x_1-x_1^{(k)})+...+\dfrac{\partial f_2(x_1^{(k)},...,x_n^{(k)})}{\partial x_n}\cdot(x_n-x_n^{(k)})=-f_2(x_1^{(k)},...,x_n^{(k)}) \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ \dfrac{\partial f_n(x_1^{(k)},...,x_n^{(k)})}{\partial x_1}\cdot(x_1-x_1^{(k)})+...+\dfrac{\partial f_n(x_1^{(k)},...,x_n^{(k)})}{\partial x_n}\cdot(x_n-x_n^{(k)})=-f_n(x_1^{(k)},...,x_n^{(k)}) \end{cases}, k=0,1,\ldots$$

Для их решения обычно используется метод Гаусса, т.к. вероятность выполнения для всех этих систем достаточного условия сходимости итераций (3.7) весьма невелика.

Алгоритм метода Ньютона, предусматривающий формирование и решение подобных систем линейных уравнений до выполнения условия

$\max\limits_{i=1,2,...,n}\{(x_i^{(k+1)}-x_i^{(k)})^2\}<\varepsilon$, много сложнее алгоритмов методов простых итераций и Зейделя, но он применяется в вычислительной практике чаще, поскольку не требует формирования и проверки выполнения условия (3.5).

3.4 Методы приближения функций

Приближением функции $f(x)$ называется ее замена многочленом вида: $P_m(x)=a_0\varphi_0(x)+a_1\varphi_1(x)+...+a_m\varphi_m(x)$, (3.8)

где $a_0, a_1,\ldots, a_m$ - постоянные коэффициенты, $\varphi_0(x)$, $\varphi_1(x),\ldots, \varphi_m(x)$ – однотипные, предварительно заданные функции, как правило непрерывно-дифференцируемые.

В вычислительной практике задачи приближения функций чаще всего решаются с целью получения аналитических функций, соответствующих зависимостям вида $y_i=f(x_i)$, $i=0,1,\ldots,n$, которые получены в результате экспериментального исследования какого-либо объекта.

Примером объекта исследования может служить емкостной реактор с мешалкой: входной переменной может быть температура в реакторе, выходной - концентрация целевого продукта, помехами - неравномерность перемешивания и обогрева.

Полученные зависимости обычно представляют в виде таблиц:

x	x_0	x_1	x_2	. . .	x_{n-1}	x_n
y	y_0	y_1	y_2	. . .	y_{n-1}	y_n

В случаях, когда наблюдается существенный разброс значений

выходной переменной, обусловленный влиянием помех, осуществляется операция ее сглаживания. Один из популярных методов сглаживания - *метод скользящего среднего*. Сглаженное значение y_i^* получается усреднением значений y_i, соответствующих значениям x_i, которые попадают в интервал усреднения $\delta x = [x_{i-z};x_{i+z}]$, где z – натуральное число, т.е.

$$y_{i+z}^* = \frac{1}{2 \cdot z + 1} \sum_{j=0}^{2 \cdot z} y_{i+j} \ .$$

Например, при $z = 1$, $y_{i+1}^* = \frac{1}{3} \sum_{j=0}^{2} y_{i+j} = \frac{1}{3}(y_i + y_{i+1} + y_{i+2})$.

Положение z экспериментальных точек в начале диапазона $[x_0;x_n]$ и стольких же в конце не изменится.

Преимущества аналитических функций перед табличными: компактность, удобство хранения, возможность дифференцирования и интегрирования. *Главная цель* приближения табличных зависимостей многочленами вида (3.8): получение значений зависимой переменной, соответствующих значениям $x_j \neq x_i$, $i = 0,1,\ldots,n$, т.е. тем, для которых эксперименты не проводились.

Получаемые в результате приближения экспериментальных зависимостей функции $y = P_m(x)$ достоверны только при $x \in [x_0;x_n]$. Попытки их использования для получения значений y, соответствующих $x < x_0$ или $x > x_n$ могут привести к серьезным ошибкам.

В вычислительной практике применяются два основных способа приближения функций: интерполяция и аппроксимация.

3.4.1 Интерполяция экспериментальных зависимостей

Постановка задачи. Пусть задан ряд значений независимой переменной $x_0 < x_1 < x_2 < \ldots < x_n$, и в результате эксперимента получены соответствующие им значения зависимой переменной y_0, $y_1,\ldots$, y_n. Сформировать многочлен $P_n(x)=a_0\varphi_0(x)+a_1\varphi_1(x)+\ldots +a_n\varphi_n(x)$, удовлетворяющий условию

$$y_i=P_n(x_i),\ i=0,1,\ldots,n, \qquad (3.9)$$

а при $x_i < x < x_{i+1}$, $i = 0,1,\ldots,n$-1 – принимающий "разумные"

значения.

Следовательно, график функции $y=P_n(x)$ должен проходить через все точки (x_i,y_i), $i = 0,1,\ldots,n$, а между этими точками - не иметь выраженных экстремумов (см. рис. 3.6).

Выбор вида функций $\varphi_i(x)$, $i=0,1,\ldots,n$ зависит от характера экспериментальной зависимости. Чаще всего на практике применяется полиномиальная интерполяция, когда $\varphi_i(x)=x^i$, $i = 0,1,\ldots,n$, т.е.

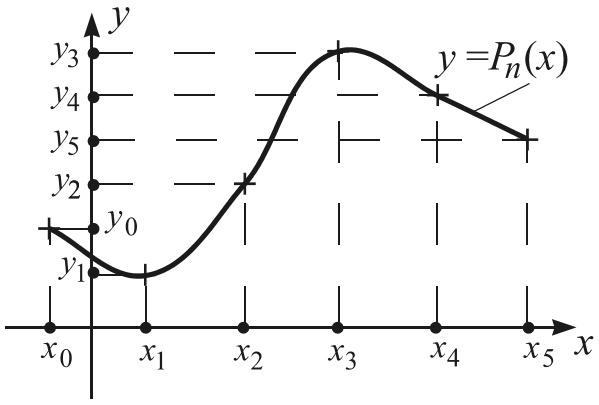


Рис. 3.6 График интерполирующего многочлена

$$P_n(x)=a_0+a_1x+a_2x^2+\ldots+a_nx^n. \quad (3.10)$$

Теоретической основой полиномиальной интерполяции является *III теорема Вейерштрасса*: любая непрерывная функция $f(x)$ на замкнутом интервале $[a;b]$ оси x может быть сколь угодно точно приближена степенным полиномом, т.е. $\forall \varepsilon > 0 \quad \exists n = n(\varepsilon): \max\limits_{x\in[a;b]} |f(x) - P_n(x)| < \varepsilon$. Степень полинома (3.10), интерполирующего зависимость $y_i = f(x_i)$, $i = 0,1,\ldots,n$, равна n, т.е. меньше числа заданных точек (x_i,y_i) на единицу.

Задача полиномиальной интерполяции сводится к определению значений коэффициентов a_0, $a_1,\ldots$, a_n полинома (3.10), при которых выполняется условие (3.9). Наиболее популярным способом ее решения является использование одной из многочисленных интерполяционных формул: Лагранжа, Ньютона, Гаусса, Бесселя, Стирлинга, Эрмита и т.д. Чаще всего на практике применяются формулы Лагранжа и Ньютона.

Определение: Полином $l_i(x)$, такой что $l_i(x_j) = \begin{cases} 1, & j = i \\ 0, & j \neq i \end{cases}$, $i,j=0,1,\ldots,n$

называется *полиномом Лагранжа*. Согласно этому определению, только при $x=x_i$ $l_i(x) = 1$, а при любом другом $l_i(x) = 0$, следовательно, полином Лагранжа может быть записан в виде: $l_i(x)=C_i\cdot(x-x_0)(x-x_1)\ldots(x-x_{i-1})(x-x_{i+1})\ldots(x-x_n)$. Значение коэффициента C_i можно определить из условия

$$l_i(x_i)=C_i\cdot(x_i-x_0)(x_i-x_1)\ldots(x_i-x_{i-1})(x_i-x_{i+1})\ldots(x_i-x_n)=1,$$

следовательно, $C_i=1/[(x_i-x_0)(x_i-x_1)\ldots(x_i-x_{i-1})(x_i-x_{i+1})\ldots(x_i-x_n)]$.

Таким образом, для заданного набора значений независимой переменной $x_0,x_1,\ldots,x_n$ полиномы Лагранжа имеют вид

$$l_i(x)=\frac{(x-x_0)(x-x_1)...(x-x_{i-1})(x-x_{i+1})...(x-x_n)}{(x_i-x_0)(x_i-x_1)...(x_i-x_{i-1})(x_i-x_{i+1})...(x_i-x_n)},\ i=0,1,\ldots,n.$$

Условие (3.9) выполняется для полиномов Лагранжа, умноженных на соответствующие значения y_i: $l_i(x_i)\cdot y_i=y_i$, i=0,1,…,n, следовательно, *формула Лагранжа* для полиномиальной интерполяции может быть записана в виде:

$$P_n(x)=\sum_{i=0}^{n}y_i l_i(x)=\sum_{i=0}^{n}y_i\frac{(x-x_0)(x-x_1)...(x-x_{i-1})(x-x_{i+1})...(x-x_n)}{(x_i-x_0)(x_i-x_1)...(x_i-x_{i-1})(x_i-x_{i+1})...(x_i-x_n)} \qquad (3.11)$$

Конечной разностью 1-го порядка для зависимости $y_i=f(x_i)$, i=0,1,…,n называется отношение разности значений зависимой переменной в двух соседних точках к разности соответствующих значений независимой переменной:

$$\Delta y_0=\frac{y_1-y_0}{x_1-x_0},\quad \Delta y_1=\frac{y_2-y_1}{x_2-x_1},\ \ldots,\quad \Delta y_{n-1}=\frac{y_n-y_{n-1}}{x_n-x_{n-1}}$$

(аналоги производной для табличной функции). Конечные разности 2-го порядка получаются из разностей 1-го порядка по правилам:

$$\Delta^2 y_0=\frac{\Delta y_1-\Delta y_0}{x_2-x_0},\ \Delta^2 y_1=\frac{\Delta y_2-\Delta y_1}{x_3-x_1},\ \ldots,\ \Delta^2 y_{n-2}=\frac{\Delta y_{n-1}-\Delta y_{n-2}}{x_n-x_{n-2}}$$

(аналоги второй производной для табличной функции). В общем виде

$$\Delta^m y_i=\frac{\Delta^{m-1}y_{i+1}-\Delta^{m-1}y_i}{x_{i+m}-x_i},\quad i=0,1,...,n-m;\quad m=1,2,...,n \qquad (3.12)$$

С ростом порядка m число конечных разностей уменьшается: при m=1 их n штук, при $m=2-(n-1)$ штук, при $m=l-(n\text{-}l+1)$ штук, при $m=n-1-2$ штуки и при $m=n$ – только одна ($\Delta^n y_0$). Интерполяционный многочлен Ньютона использует конечные разности рассматриваемой табличной функции и существует в двух формах.

Первая (левая) формула Ньютона имеет вид:

$P_n(x)=a_0+a_1(x\text{-}x_0)+a_2(x\text{-}x_0)(x\text{-}x_1)+\ldots+a_n(x\text{-}x_0)(x\text{-}x_1)\ldots(x\text{-}x_{n\text{-}1})$

(крайнее правое значение $x=x_n$ в формуле не используется). Значения коэффициентов a_0, a_1,…, a_n определяются из условия (3.9):

$$P_n(x_0)=y_0=a_0\ \rightarrow\ a_0=y_0,\quad P_n(x_1)=y_1=y_0+a_1(x_1-x_0)\ \rightarrow\ a_1=\frac{y_1-y_0}{x_1-x_0}=\Delta y_0.$$

Аналогично можно показать, что $a_2=\Delta^2 y_0$, $a_3=\Delta^3 y_0,\ldots$, $a_n=\Delta^n y_0$, следовательно,

$$P_n(x) = y_0 + \sum_{k=1}^{n}(x - x_0)(x - x_1)\ldots(x - x_{k-1})\Delta^k y_0. \qquad (3.13)$$

Вторая (правая) формула Ньютона:

$P_n(x)=a_0+a_1(x-x_n)+a_2(x-x_n)(x-x_{n-1})+\ldots+a_n(x-x_n)(x-x_{n-1})\ldots(x-x_1)$

(не используется крайнее левое значение $x = x_0$). Из условия (3.9) в данном случае получается $a_0=y_n$, $a_1=\Delta y_{n-1}$, $a_2=\Delta^2 y_{n-2},\ldots$, $a_n=\Delta^n y_0$, т.е.

$$P_n(x) = y_n + \sum_{k=1}^{n}(x - x_n)(x - x_{n-1})\ldots(x - x_{n-k+1})\Delta^k y_{n-k}. \qquad (3.14)$$

Пример. Интерполировать зависимость $x_0=0$; $x_1=1$; $x_2=2$; $x_3=3$;
$y_0=-1$; $y_1=0$; $y_2=7$; $y_3=26$;

Конечные разности:

$$\Delta y_0 = \frac{y_1 - y_0}{x_1 - x_0} = -1,\ \ \Delta y_1 = \frac{y_2 - y_1}{x_2 - x_1} = 7,\ \ \Delta y_2 = \frac{y_3 - y_2}{x_3 - x_2} = 19;$$

$$\Delta^2 y_0 = \frac{\Delta y_1 - \Delta y_0}{x_2 - x_0} = 3,\quad \Delta^2 y_1 = \frac{\Delta y_2 - \Delta y_1}{x_3 - x_1} = 6;\quad \Delta^3 y_0 = \frac{\Delta^2 y_1 - \Delta^2 y_0}{x_3 - x_0} = 1.$$

Левая формула:

$$P_3(x)=y_0+\Delta y_0(x-x_0)+\Delta^2 y_0(x-x_0)(x-x_1)+\Delta^3 y_0(x-x_0)(x-x_1)(x-x_2)=$$
$$= -1+1\cdot(x-0)+3\cdot(x-0)\cdot(x-1)+1\cdot(x-0)\cdot(x-1)\cdot(x-2) = x^3-1.$$

Правая формула:

$$P_3(x)=y_3+\Delta y_2(x-x_3)+\Delta^2 y_1(x-x_3)(x-x_2)+\Delta^3 y_0(x-x_3)(x-x_2)(x-x_1)=$$
$$=26+19\cdot(x-3)+6\cdot(x-3)\cdot(x-2)+1\cdot(x-3)\cdot(x-2)\cdot(x-1) = x^3-1.$$

Существует единственный полином степени $\le n$, интерполирующий заданные точки (x_i,y_i), $i=0,1,\ldots,n$. Следовательно, для одной и той же зависимости $y_i=f(x_i)$, $i=0,1,\ldots,n$ все интерполяционные формулы порождают один и тот же полином. Разница между различными интерполяционными формулами заключается в способе формирования полинома вида (3.10).

В отличие от формулы Лагранжа, которая имеет одинаковую трудоемкость при любом расположении точек (x_i,y_i), формулы Ньютона менее трудоемки, если эти точки являются равноотстоящими,

т.е. $\Delta x_i = x_i - x_{i-1} = const$, $i=1,2,\ldots,n$ (конечные разности вычисляются проще, поскольку их знаменатели известны заранее). Кроме того, при увеличении числа точек (x_i,y_i) использование формул Ньютона потребует лишь добавления к уже сформированному многочлену $P_n(x)$ дополнительных слагаемых, а использование формулы Лагранжа – повторения операции его формирования. С другой стороны, применение формул Ньютона требует предварительного вычисления всех конечных разностей интерполируемой зависимости $y_i = f(x_i)$, $i=0,1,\ldots,n$. Для удобства их вычисления рекомендуется формировать следующие таблицы:

x	y	Δy	$\Delta^2 y$	$\Delta^3 y$	. . . . .	$\Delta^{n-1} y$	$\Delta^n y$
x_0	$\boldsymbol{y_0}$	$\boldsymbol{\Delta y_0}$	$\boldsymbol{\Delta^2 y_0}$	$\boldsymbol{\Delta^3 y_0}$	. . . . .	$\boldsymbol{\Delta^{n-1} y_0}$	$\boldsymbol{\Delta^n y_0}$
x_1	y_1	Δy_1	$\Delta^2 y_1$	. . . . .	. . . . .	$\boldsymbol{\Delta^{n-1} y_1}$	0
x_2	y_2	Δy_2	. . . . .	$\Delta^3 y_{n-4}$	. . . . .	0	. . . . .
x_3	y_3	. . . . .	$\Delta^2 y_{n-3}$	$\boldsymbol{\Delta^3 y_{n-3}}$	. . . . .	. . . . .	. . . . .
. . . . .	. . . . .	Δy_{n-2}	$\boldsymbol{\Delta^2 y_{n-2}}$	0	. . . . .	. . . . .	. . . . .
x_{n-1}	y_{n-1}	$\boldsymbol{\Delta y_{n-1}}$	0	0	. . . . .	. . . . .	. . . . .
x_n	$\boldsymbol{y_n}$	0	0	0	. . . . .	0	0

Коэффициенты левой формулы Ньютона стоят в верхней строке таблицы, коэффициенты правой - на диагонали.

На практике полиномиальная интерполяция с использованием формул Лагранжа, Ньютона и им подобных применяется при $n \le 5\ldots6$. При большем числе точек (x_i,y_i) ее результаты становятся малопригодными: получаемые полиномы $P_n(x)$ удовлетворяют условию $P_n(x_i) = y_i$, $i = 0,1,\ldots,n$, но в промежутках между точками (x_i,y_i) могут принимать явно "неразумные", недостоверные значения. Поэтому при $n > 5\ldots6$ осуществляют кусочную полиномиальную интерполяцию, т.е. применяют формулы (3.12) - (3.14) не ко всему отрезку $[x_0;x_n]$, а последовательно к его частям, содержащим не более 5…6 точек (x_i,y_i). Наиболее популярны кусочно-линейная и кусочно-квадратичная интерполяция.

Кусочно-линейная интерполяция зависимости $y_i = f(x_i)$, $i = 0,1,\ldots,n$ предусматривает соединение каждой пары соседних точек (x_i,y_i) отрезком прямой линии, т.е. формирование для каждого отрезка $[x_i,x_{i+1}]$ полинома $P_1^{(i)}(x)$, $i=0,1,\ldots n-1$, например, по формуле Лагранжа $P_1^{(i)}(x)=y_i\frac{x-x_{i+1}}{x_i-x_{i+1}}+y_{i+1}\frac{x-x_i}{x_{i+1}-x_i}$,

а по левой формуле Ньютона $P_1^{(i)}(x)=y_i+\frac{y_{i+1}-y_i}{x_{i+1}-x_i}\cdot(x-x_i)$.

Кусочно-квадратичная интерполяция сводится к формированию для каждого отрезка $[x_i,x_{i+2}]$ полинома $P_2^{(i)}(x)$, $i = 0,2,\ldots n-2$, т.е. предусматривает соединение каждой тройки соседних точек (x_i,y_i) отрезком квадратичной параболы. По формуле Лагранжа

$$P_2^{(i)}(x)=y_i\frac{(x-x_{i+1})(x-x_{i+2})}{(x_i-x_{i+1})(x_i-x_{i+2})}+y_{i+1}\frac{(x-x_i)(x-x_{i+2})}{(x_{i+1}-x_i)(x_{i+1}-x_{i+2})}+y_{i+2}\frac{(x-x_i)(x-x_{i+1})}{(x_{i+2}-x_{i+1})(x_{i+2}-x_{i+2})},$$

по правой формуле Ньютона

$$P_2^{(i)}(x)=y_{i+2}+\frac{y_{i+2}-y_{i+1}}{x_{i+2}-x_{i+1}}(x-x_{i+2})+\frac{\frac{y_{i+2}-y_{i+1}}{x_{i+2}-x_{i+1}}-\frac{y_{i+1}-y_i}{x_{i+1}-x_i}}{x_{i+2}-x_i}(x-x_{i+2})(x-x_{i+1}).$$

Рассмотренные способы кусочной интерполяция просты, их результаты вполне надежны. Единственный недостаток - интерполирующая функция не гладкая (ломаная), т.е. ее производная может иметь разрывы в точках (x_i,y_i). Это существенно в случаях, когда зависимой переменной является первообразная от действительно интересующей исследователя величины (например, нас интересует изменение скорости тела во времени, а в эксперименте снята зависимость его пути от времени). Существует способ кусочной интерполяции, лишенный этого недостатка - *сплайн-интерполяция*.

Английское слово *сплайн* обозначает гибкую рейку из упругого материала. Цепляя к сплайну пружины разной жесткости и грузила разного веса, можно получить кривую, интерполирующую заданное множество точек (x_i,y_i), $i = 0,1,\ldots,n$. Сплайн не разрушается, т.е. образует гладкую кривую $S(x)$. В теории балок доказывается, что $S(x)$ на каждом отрезке $[x_{i-1};x_i]$, $i=1,2,\ldots,n$ представляет собой кубический полином, причем соседние полиномы, их первые и вторые производные соединяются непрерывно. Поэтому $S(x)$ называется кубическим сплайном.

Полином, формируемый для каждого из отрезков $[x_{i-1};x_i]$, $i=1,2,\ldots,n$ имеет вид: $a_i+b_i(x-x_{i-1})+c_i(x-x_{i-1})^2+d_i(x-x_{i-1})^3$. Коэффициенты этих полиномов определяются из следующих соотношений:

$$a_i = y_{i-1},\ i = 1,2,\ldots,n; \tag{3.15}$$

$$c_i(x_i-x_{i-1})+2c_{i+1}(x_{i+1}-x_{i-1})+c_{i+2}(x_{i+1}-x_i)=3\left(\frac{y_{i+1}-y_i}{x_{i+1}-x_i}-\frac{y_i-y_{i-1}}{x_i-x_{i-1}}\right), \tag{3.16}$$

$$i=1,\ldots,n-1 \qquad c_1=c_{n+1}=0;$$

$$b_i = \frac{y_i - y_{i-1}}{x_i - x_{i-1}} - \frac{x_i - x_{i-1}}{3}(c_{i+1} + 2 \cdot c_i),\ i = 1,2,...,n\ ; \qquad (3.17)$$

$$d_i = \frac{c_{i+1} - c_i}{3(x_i - x_{i-1})},\ i = 1,2,...,n\ ; \qquad (3.18)$$

т.е. значения коэффициентов b_i, d_i, i = 1,2,…,n зависят от значений коэффициентов c_i, для определения которых необходимо решить систему (3.16) линейных уравнений порядка (n-1). Системы вида (3.16) обычно хорошо обусловлены, для них всегда выполняется условие $|a_{ii}| > \sum_{j=1, j\neq i}^{n} |a_{ij}|$, i = 1,2,...,n, поэтому их можно без преобразований решать методами Гаусса, Якоби, Зейделя.

Интерполяция не находит широкого применения для приближения экспериментальных зависимостей, т.к. нет смысла получать функцию, график которой проходит точно через экспериментальные точки, если эти точки несут в себе погрешность. Среднее значение погрешности эксперимента обычно известно, поэтому значительно чаще в инженерной практике экспериментальные зависимости аппроксимируют, т.е. приближают функциями, графики которых проходят достаточно близко к точкам (x_i,y_i), i = 0,1,…,n

3.4.2 Аппроксимация экспериментальных зависимостей

Задача построения многочлена вида (3.8), значения которого в точках $x_i, i=0,1,...,n$ в достаточной степени соответствуют значениям y_i, i = 0,1,…,n, называется задачей аппроксимации зависимости $y_i = f(x_i)$ (рис.3.7).

Постановка задачи: Подобрать элементарные функции $\varphi_j(x)$: x^j, $\sin(\beta_j x)$, $e^{\gamma_j x}$ и т.п., порядок m и определить значения коэффициентов многочлена $P_m(x) = a_0\varphi_0(x) + a_1\varphi_1(x) + ... + a_m\varphi_m(x),$ при которых он достаточно точно соответствует исходной экспериментальной зависимости.

Чаще всего в вычислительной практике используется полиномиальная аппроксимация, когда

$\varphi_j(x) = x^j,\ j = 0,1,...,m,$ т.е. $P_m(x) = a_0 + a_1x + a_2x^2 + ... + a_mx^m.$ Наиболее популярными методами полиномиальной аппроксимации являются метод наименьших квадратов и метод ортогональных полиномов Чебышева.

Метод наименьших квадратов предусматривает произвольный выбор порядка m полинома $P_m(x)$ и определение значений коэффициентов $a_0, a_1, ..., a_m$ из условия

$$\delta = \sqrt{\frac{1}{n+1} \cdot \sum_{i=0}^{n} (P_m(x_i) - y_i)^2} \to \min, \qquad (3.19)$$

т.е. минимума среднеквадратичного отклонения $P_m(x_i)$ от y_i, i=0,1,…,n.

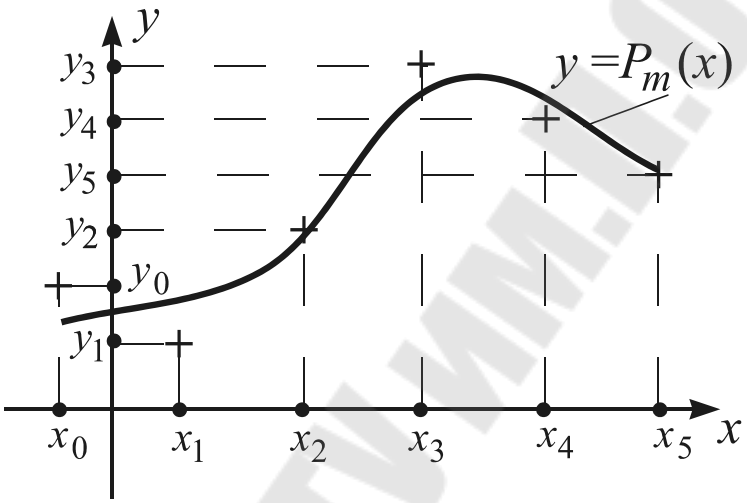


Рис. 3.7 График аппроксимирующего многочлена

При выбранном m задача определения значений коэффициентов a_0, a_1,..., a_m сводится к поиску минимума функции

$$S(a_0, a_1, ..., a_m) = \sum_{i=0}^{n} (a_0 + a_1 x_i + a_2 x_i^2 + ... + a_m x_i^m - y_i)^2.$$

Как известно из курса высшей математики, функция S достигнет минимума в точке, где

$\frac{\partial S}{\partial a} = 2 \cdot \sum_{i=0}^{n} (a_0 + a_1 x_i + a_2 x_i^2 + ... + a_m x_i^m - y_i) \cdot x_i^j = 0, \quad j = 0,1,...,m$. Преобразуя эти выражения, получим систему линейных уравнений порядка $(m+1)$ с неизвестными a_0, a_1 ,..., a_m:

$$\begin{cases} a_0(n+1) + a_1 \sum_{i=0}^{n} x_i + a_2 \sum_{i=0}^{n} x_i^2 + ... + a_m \sum_{i=0}^{n} x_i^m = \sum_{i=0}^{n} y_i; \\ a_0 \sum_{i=0}^{n} x_i + a_1 \sum_{i=0}^{n} x_i^2 + a_2 \sum_{i=0}^{n} x_i^3 + ... + a_m \sum_{i=0}^{n} x_i^{m+1} = \sum_{i=0}^{n} x_i y_i; \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ a_0 \sum_{i=0}^{n} x_i^m + a_1 \sum_{i=0}^{n} x_i^{m+1} + a_2 \sum_{i=0}^{n} x_i^{m+2} + ... + a_m \sum_{i=0}^{n} x^{2m} = \sum_{i=0}^{n} x_i^m y_i; \end{cases} \qquad (3.20)$$

Доказано, что определитель системы (3.20) отличен от нуля: ее решение существует и единственно. Диагональные коэффициенты матрицы системы всегда отличны от нуля, т.е. ее можно без преобразований решать методом Гаусса. Применить итерационные методы, как правило, не удается.

Довольно популярным способом выявления степени соответствия полученного в результате полинома $y=P_m(x)$ зависимости $y_i=f(x_i)$, $i = 0,1,...n$ является вычисление значения критерия Фишера $F=\frac{\sigma_y}{\sigma_{\text{ост}}}$, где $\sigma_y=\frac{1}{n}\cdot\sum_{i=0}^{n}(y_i-\bar{y})^2$ - дисперсия относительно среднего значения y_i, $i = 0,...,n$: $\bar{y}=\frac{1}{n+1}\cdot\sum_{i=0}^{n}y_i$; $\sigma_{\text{ост}}=\frac{1}{n-m}\cdot\sum_{i=0}^{n}[P_m(x_i)-y_i]^2$ - остаточная дисперсия. Критерий Фишера показывает, во сколько раз рассеяние исходной экспериментальной зависимости относительно полученного полинома меньше, чем ее рассеяние относительно среднего арифметического значения y_i, $i = 0,1,...,n$.

Степень соответствия тем выше, чем больше полученное значение F превышает табличное $F_p(f_1,f_2)$ для выбранного уровня значимости p и чисел степеней свободы f_1, f_2. Уровень значимости p – это число, полученное вычитанием из единицы значения вероятности того, что $P_m(x) - p \le y \le P_m(x) + p$, (например $p = 0.05$, если вероятность равна 0.95), $f_1=n$, $f_2= n - m$. Если полученное значение F окажется меньше табличного, необходимо изменить условия формирования полинома $y = P_m(x)$:

а) увеличить его порядок m (с учетом ограничения $m < n$, т.к. при $m = n$ $P_m(x)$ станет интерполяционным полиномом);

б) выбрать другие элементарные функции φ_j, $j = 0,1,..., m$;

в) увеличить уровень значимости p;

г) предварительно сгладить зависимость $y_i = f(x_i)$, $i = 0,1,...,n$.

На практике метод наименьших квадратов применяют лишь при $m \le 5\ldots7$, т.к. при больших m система (3.20) становится плохо обусловленной и коэффициенты $a_0, a_1, ..., a_m$ определяются с большими ошибками. Еще один существенный недостаток этого метода – необходимость повторного решения задачи в случаях, когда

точность первоначального решения недостаточна.

Метод ортогональных полиномов Чебышева лишен указанных недостатков. Многочлен $P_m(x)$ при его использовании также имеет вид (3.8), но функции $\varphi_j(x),\ j = 1,2,...,\ m$ - это полиномы, удовлетворяющие условиям,

$$\begin{cases} \sum_{i=0}^{n} \varphi_j(x_i) \cdot \varphi_k(x_i) = 0, \ j \neq k. \\ \sum_{i=0}^{n} [\varphi_j(x_i)]^2 \neq 0, \ \ j = 0,1,2,...,m. \end{cases} \tag{3.21}$$

$\varphi_0(x) = 1.$ $\varphi_1(x)$ можно получить из условий (3.21), положив $\varphi_1(x) = x + \alpha$.

$$\sum_{i=0}^{n} \varphi_0(x_i) \cdot \varphi_1(x_i) = 0 \Rightarrow \sum_{i=0}^{n} (x_i + \alpha) = 0 \Rightarrow \alpha = -\frac{1}{n+1} \cdot \sum_{i=0}^{n} x_i, \ \varphi_1(x_i) = x_i - \frac{1}{n+1} \cdot \sum_{i=0}^{n} x_i \ .$$

Следующий полином Чебышева вычисляется по двум предыдущим:

$$\varphi_{r+1}(x) = (x + \beta_{r+1}) \cdot \varphi_r(x) + \gamma_{r+1} \cdot \varphi_{r-1}(x), \quad r = 1,2,...,\ m-1,$$

где $\beta_{r+1} = -\sum_{i=0}^{n} x_i \cdot [\varphi_r(x_i)]^2 \Big/ \sum_{i=0}^{n} [\varphi_r(x_i)]^2$, $\gamma_{r+1} = -\sum_{i=0}^{n} x_i \cdot \varphi_r(x_i) \cdot \varphi_{r-1}(x_i) / \sum_{i=0}^{n} [\varphi_{r-1}(x_i)]^2$.

Например, $\varphi_2(x) = (x + \beta_2) \cdot \varphi_1(x) + \gamma_2 \cdot \varphi_0(x),$ где

$$\beta_2 = -\sum_{i=0}^{n} x_i \cdot \left[x_i - \frac{1}{n+1} \cdot \sum_{i=0}^{n} x_i \right]^2 \Big/ \sum_{i=0}^{n} \left[x_i - \frac{1}{n+1} \cdot \sum_{i=0}^{n} x_i \right]^2, \gamma_2 = -\sum_{i=0}^{n} x_i \cdot \left[x_i - \frac{1}{n+1} \cdot \sum_{i=0}^{n} x_i \right].$$

Для получения конкретного полинома при фиксированном *m* необходимо найти по этим формулам полиномы $\varphi_j(x),\ j = 1,2,...,\ m$ и определить значения коэффициентов $a_0, a_1, ..., a_m$. Доказано, что наиболее вероятные значения коэффициентов равны

$$a_r = \sum_{i=0}^{n} y_i \cdot \varphi_r(x_i) / \sum_{i=0}^{n} [\varphi_r(x_i)]^2, \quad r = 0,1,2,...,m.$$

Использование метода ортогональных полиномов не связано с решением систем линейных уравнений и позволяет легко переходить от полинома $P_m(x)$ к полиному $P_{m+1}(x)$ при недостаточной точности первого. Для этого необходимо сформировать полином $\varphi_{m+1}(x)$, определить значение коэффициента a_{m+1} и добавить произведение $a_{m+1} \cdot \varphi_{m+1}(x)$ к многочлену $P_m(x)$. Поэтому, хотя алгоритм этого метода значительно сложнее алгоритма метода наименьших

квадратов, в вычислительной практике чаще применяется метод ортогональных полиномов.

3.5 Формулы численного интегрирования

Численное интегрирование – это приближенное вычисление определенных интегралов вида

$$I = \int_a^b f(x)\ \ dx. \qquad (3.22)$$

Этот способ вычисления интегралов применяется, когда первообразную функции $f(x)$ определить сложно или невозможно (например, $f(x)$ задана таблицей значений).

Постановка задачи: Вычислить интеграл (3.22) с заданной степенью точности по значениям подынтегральной функции $f(x)$ в некоторых точках отрезка $[a;b]$ оси x.

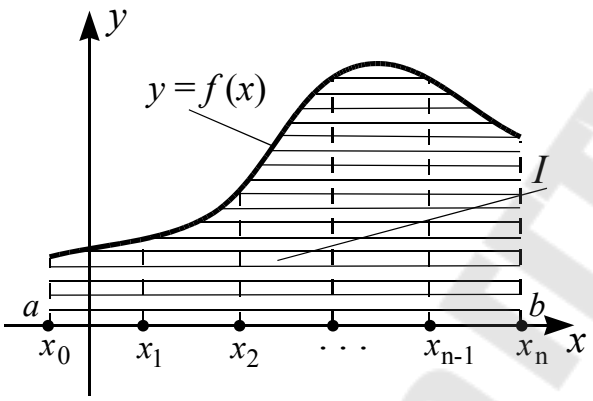


Рис. 3.8 Схема численного интегрирования

Определение: Упомянутые точки отрезка $[a;b]$ называют *узлами*, а проходящие через них перпендикуляры к оси x - *ординатами*.

Общая схема численного интегрирования. Если значения a и b конечны, а функция $f(x)$ непрерывна на отрезке $[a;b]$, то интеграл I есть площадь плоской фигуры, ограниченной кривой $y=f(x)$, осью x и ординатами $x=a$, $x=b$. Приближенное вычисление интеграла сводится к разбиению отрезка $[a;b]$ на множество более мелких отрезков, приближенному определению площади каждой получившейся криволинейной трапеции и их суммированию (рис.1.8).

Определение: В литературе численное определение интеграла вида (3.22) называют *квадратурой*, а формулы численного интегрирования – *квадратурными*.

Имеется две разновидности квадратурных формул. Первая предусматривает разбиение отрезка $[a;b]$ на равные микроотрезки $[x_{i-1};x_i]$, $i=1,2,...,n$ $(x_0=a, x_n=b)$ длиной $h=(b-a)/n$. Наиболее популярные

формулы этой разновидности – формулы прямоугольников, трапеций и Симпсона. Формулы второй разновидности основаны на определении положения точек x_i, i=1,2,...,n-1 внутри отрезка [a;b], позволяющего достичь максимальной точности вычисления интеграла при заданном числе точек. Из этой разновидности чаще всего применяется формула Гаусса.

При использовании формул первой группы $I = \int_a^b f(x)\ dx = \sum_{i=1}^{n} \int_{x_{i-1}}^{x_i} f(x)\ dx$.

Приближенные значения интегралов $I_i = \int_{x_{i-1}}^{x_i} f(x)\ dx$, i=1,2,...,n определяются путем замены подынтегральной функции f (x) элементарной функцией внутри всех отрезков [x_{i-1};x_i].

Формула прямоугольников (рис.3.9). На каждом отрезке [x_{i-1};x_i], i=1,2,...,n площадь под кривой $y=f$ (x) принимается приближенно равной площади прямоугольника с основанием h и высотой $f\left(\frac{x_{i-1}+x_i}{2}\right)$,

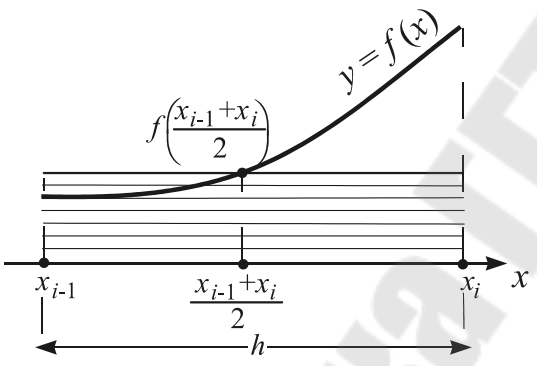


Рис. 3.9 Формула прямоугольников

т.е. $I_i \approx h \cdot f\left(\frac{x_{i-1}+x_i}{2}\right)$, i=1,2,...,n, а

$$I \approx h \cdot \sum_{i=1}^{n} f\left(\frac{x_{i-1}+x_i}{2}\right). \qquad (3.23)$$

Формула трапеций (рис. 3.10). На отрезках [x_{i-1};x_i], i=1,2,...,n площадь под кривой $y=f$ (x) заменяется площадью трапеции с основанием h и высотами f (x_{i-1}) и f (x_i). Следовательно $I_i \approx \frac{h}{2} \cdot [f(x_{i-1}) + f(x_i)]$, i=1,2…,n, т.е.

$$I \approx \frac{h}{2} \cdot \sum_{i=1}^{n} [f(x_{i-1}) + f(x_i)]. \qquad (3.24)$$

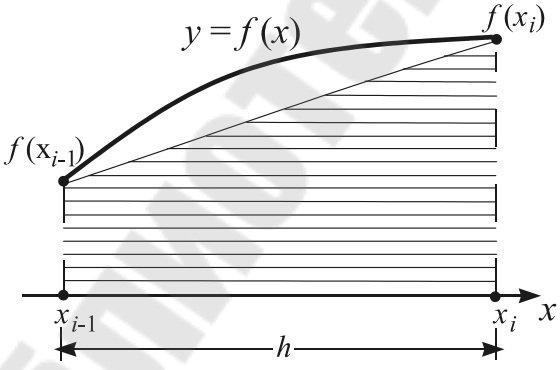


Рис. 3.10 Формула трапеций

Формула Симпсона (рис.3.11). На отрезке [x_{i-1};x_i], i=1,2,...,n кривая $y=f$ (x) заменяется квадратичной параболой, график которой проходит через точки

$[x_{i-1}, f(x_{i-1})]$, $\left[\left(\frac{x_{i-1}+x_i}{2}\right), f\left(\frac{x_{i-1}+x_i}{2}\right)\right]$, $[x_i, f(x_i)]$. По формуле Лагранжа уравнение этой параболы можно представить в виде

$$P_2(x) = \frac{(x-z_i)(x-x_i)}{(x_{i-1}-z_i)(x_{i-1}-x_i)} \cdot f(x_{i-1}) + \frac{(x-x_{i-1})(x-x_i)}{(z_i-x_{i-1})(z_i-x_i)} \cdot f(z_i) +$$
$$+ \frac{(x-x_{i-1})(x-z_i)}{(x_i-x_{i-1})(x_i-z_i)} \cdot f(x_i),$$

где $z_i=(x_{i-1}+x_i)/2$. Учитывая, что $x_i-x_{i-1}=h$, а $z_i-x_{i-1}=x_i-z_i=h/2$,

$$I_i \approx 2 \cdot \frac{f(x_{i-1})}{h^2} \cdot \int_{x_{i-1}}^{x_i} (x-z_i)(x-x_i)\,dx - 4 \cdot \frac{f(z_i)}{h^2} \cdot \int_{x_{i-1}}^{x_i} (x-x_{i-1})(x-x_i)\,dx +$$
$$+ 2 \cdot \frac{f(x_i)}{h^2} \cdot \int_{x_{i-1}}^{x_i} (x-x_{i-1})(x-z_i)\,dx,\ i=1,2,...,n.$$

Взяв все три интеграла по частям, получим $I_i \approx \frac{h}{6} \cdot [f(x_{i-1}) + 4 \cdot f(z_i) + f(x_i)]$, $i=1,2,...,n$, следовательно

$$I \approx \frac{h}{6} \cdot \sum_{i=1}^{n} \left[f(x_{i-1}) + 4 \cdot f\left(\frac{x_{i-1}+x_i}{2}\right) + f(x_i) \right]. \qquad (3.25)$$

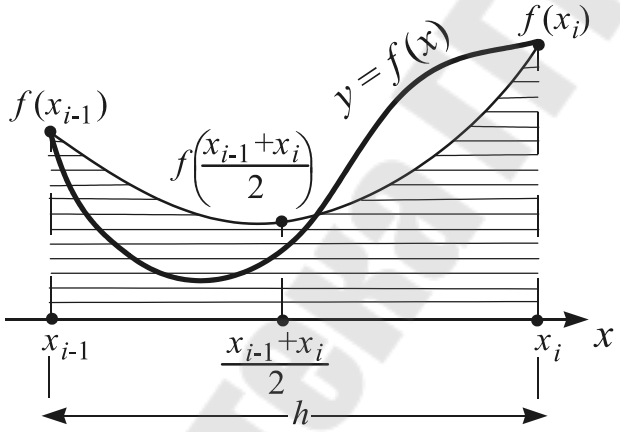


Рис. 3.11 Формула Симпсона

Формулу Симпсона называют иногда *формулой парабол*.

Заменяя $f(x)$ на отрезках $[x_{i-1};x_i]$, $i=1,2,...,n$ полиномами 3-го, 4-го и т.д. порядков можно получить более сложные квадратурные формулы, носящие общее название: *формулы Ньютона-Котеса*.

Формула Симпсона дает абсолютно точный результат при интегрировании *не только квадратичных, но и кубических* полиномов для любого числа n. Эта формула соединяет в себе достоинства простоты и высокой точности. Для более простых формул прямоугольников и трапеций характерна гораздо меньшая точность.

Ошибки вычисления интеграла (3.22) при фиксированном числе

микроотрезков и можно вычислить по формулам:

$$O_p = \frac{(b-a)^3}{24n^2}\cdot f^{II}(c),\ O_t = -\frac{(b-a)^3}{12n^2}\cdot f^{II}(c),\ O_s = -\frac{(b-a)^5}{180\ n^4}\cdot f^{IV}(c),\quad a \le c \le b \qquad (3.26)$$

Из этих формул видно, что при одинаковом числе n формула прямоугольников вдвое точнее формулы трапеций, а формула Симпсона точнее обеих на порядок. Применять формулы (3.26) в вычислительной практике неудобно из-за необходимости вычисления производных функции $f(x)$.

Основой алгоритма вычисления интегралов вида (3.22) по формулам (3.23)-(3.25) является прием последовательного удвоения числа микроотрезков n и сравнения получаемых значений интеграла. Согласно формулам (3.26), удвоение числа n приводит к уменьшению O_p и O_t в 4 раза , а O_s - в 16 раз. Начальное число n обычно равно 10-20. Процесс удвоения n и пересчета значения интеграла заканчивают при выполнении неравенства: $\left|I^{(n)} - I^{(2n)}\right| < \varepsilon \cdot K_F$, где ε - заданная точность (обычно $10^{-4}\ldots10^{-5}$), K_F - коэффициент, зависящий от используемой формулы (K_p=K_t=3, K_s=16). Увеличивать точность не имеет смысла, т.к. с ростом n и уменьшением ошибки ограничения увеличивается ошибка округления (при n >700 для формул прямоугольников и трапеций, n >100 для формулы Симпсона общая ошибка начинает увеличиваться).

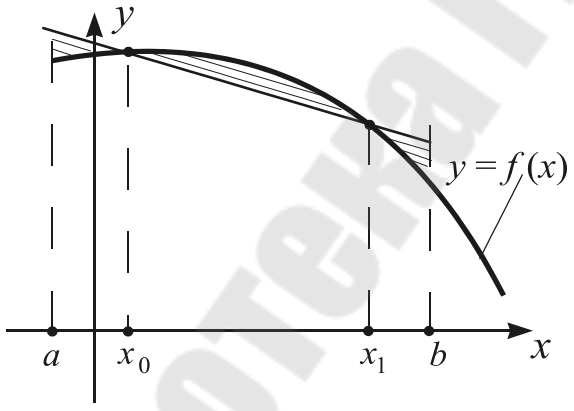


Рис. 3.12 Иллюстрация к формуле Гаусса

Квадратурная формула Гаусса. Основной принцип квадратурных формул второй разновидности виден из рис. 3.12: необходимо так разместить точки x_0 и x_1 внутри отрезка [a;b], чтобы площади "треугольников" в сумме были равны площади "сегмента". При использовании формулы Гаусса исходный отрезок [a;b] сводится к отрезку [-1;1] заменой переменной x на $0.5\cdot(b-a)\cdot t + 0.5\cdot(b+a)$.

Тогда $I = \int_a^b f(x)dx = \int_{-1}^{1} \varphi(t)dt$, где $\varphi(t) = \frac{b-a}{2}\cdot f\left(\frac{b-a}{2}\cdot t + \frac{b+a}{2}\right)$.

Такая замена возможна, если a и b конечны, а функция $f(x)$

непрерывна на [a;b]. Формула Гаусса при n точках x_i, i=0,1,..,n-1 внутри отрезка [a;b]:

$$I \approx \sum_{n=0}^{n-1} [A_i \cdot \varphi(t_i)], \qquad (3.27)$$

где t_i и A_i для различных n приводятся в справочниках. Например, при n=2 $t_0 = -1/\sqrt{3},\ t_1 = 1/\sqrt{3},$ A_0=A_1=1; при n=3: t_0=t_2≈0.775, t_1=0, A_0=A_2≈0.555, A_1≈0.889.

Алгоритм вычисления интеграла (3.22) по формуле Гаусса предусматривает не удвоение числа микроотрезков, а увеличение числа ординат на 1 и сравнение полученных значений интеграла. Преимущество формулы Гаусса – высокая точность при сравнительно малом числе ординат. Недостатки: неудобна при расчетах вручную; необходимо держать в памяти ЭВМ значения t_i, A_i для различных n.

3.6 Решение обыкновенных дифференциальных уравнений

Обыкновенные дифференциальные уравнения - это уравнения, содержащие производные функций одного переменного. Они используются для математического описания физических ситуаций, требующих рассмотрения *степени изменения* одной переменной величины по отношению к другой: уравнения химической кинетики, теплопроводности, диффузии и т.д. Необходимость решения таких уравнений часто возникает в вычислительной практике.

Аналитические методы решения обыкновенных дифференциальных уравнений редко удается использовать для решения практических задач. Задолго до появления ЭВМ для этого использовали численные методы.

Постановка задачи: Найти решение y=y(x) уравнения y'=f(x, y), удовлетворяющее условию y(x_0)=y_0. Условие y(x_0)=y_0 называется начальным условием и используется для выделения одной конкретной интегральной кривой y(x) из множества решений уравнения y'=f (x, y). Например, для того, чтобы выделить из множества решений уравнения y'=y решение y=e^x необходимо задать условие y(0)=1. Эта задача называется *задачей Коши*. На практике она обычно решается для заданного интервала значений аргумента [x_0; x_n].

3.6.1 Методы численного решения задачи Коши для одного уравнения

Решение задачи Коши можно представить в виде $y(x) = \int_{x_0}^{x} f(x,y)\,dx + c$. Поскольку $y(x_0) = y_0$, $y_0 = \int_{x_0}^{x_0} f(x,y)\,dx + c \Rightarrow c = y_0$ и $y(x) = y_0 + \int_{x_0}^{x} f(x,y)\,dx$. Разбив отрезок $[x_0;\ x_n]$ точками $x_1, x_2 .. x_{n-1}$ на n равных микроотрезков $[x_i;\ x_{i+1}]$, $i=0,1,\ldots,n-1$ длиной $h=(x_n - x_0)/n$, запишем правило определения соответствующих значений $y_1, y_2, \ldots, y_n$:

$$y_1 = y_0 + \int_{x_0}^{x_1} f(x,y)\,dx;$$

$$y_2 = y_0 + \int_{x_0}^{x_2} f(x,y)\,dx = y_0 + \int_{x_0}^{x_1} f(x,y)\,dx + \int_{x_1}^{x_2} f(x,y)\,dx = y_1 + \int_{x_1}^{x_2} f(x,y)\,dx;$$

$$y_n = y_0 + \int_{x_0}^{x_n} f(x,y)\,dx = y_0 + \int_{x_1}^{x_2} f(x,y)\,dx + \ldots + \int_{x_{n-1}}^{x_n} f(x,y)\,dx = y_{n-1} + \int_{x_{n-1}}^{x_n} f(x,y)\,dx.$$

В общем виде

$$y_{i+1} = y_i + \int_{x_i}^{x_{i+1}} f(x,y)\,dx, \quad i = 0,1,..n-1 \qquad (3.28)$$

Используя для приближенного вычисления интегралов $\int_{x_i}^{x_{i+1}} f(x,y)\,dx$ различные квадратурные формулы, можно получить различные методы численного решения задачи Коши. Наиболее популярны метод Эйлера и его модификации, метод Рунге-Кутта. Для достижения необходимой точности решения используется прием уменьшения длины микроотрезков, повторения вычислений и сравнения полученных результатов.

Классический метод Эйлера использует для вычисления интегралов $\int_{x_i}^{x_{i+1}} f(x,y)\,dx$ "формулу левых прямоугольников" (высота прямоугольника принимается равной значению подынтегральной функции на левой границе микроотрезка $[x_i;\ x_{i+1}]$):

$$\int_{x_i}^{x_{i+1}} f(x)\,dx \approx h \cdot f(x_i) \text{ т.е. } y_{i+1} = y_i + h \cdot f(x_i, y_i),\ i = 0,\ldots,n-1. \qquad (3.29)$$

Геометрическая интерпретация (рис. 3.13): точки (x_{i+1}, y_{i+1}), $i=0,1,\ldots,n-1$ – это точки пересечения касательных к кривой $y(x)$ в точках (x_i, y_i) и прямых $x = x_{i+1}$.

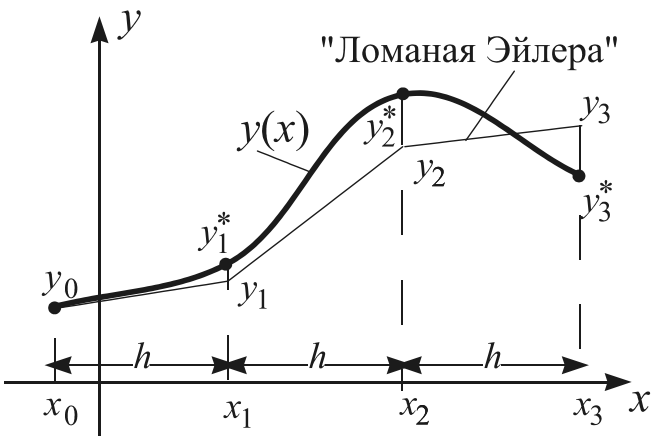


Рис.3.13 Иллюстрация к методу Эйлера

Ошибки, возникающие при определении значений $y_1, y_2, \ldots, y_n$, приводят к тому, что каждая следующая касательная проводится к какой-то другой интегральной кривой из семейства решений уравнения. Такое свойство метода называют накоплением ошибки.

Метод прост, но имеет весьма малую точность, его называют методом первого порядка, так как его основное соотношение совпадает с разложением функции $y=y(x)$ в ряд Тейлора в окрестности точки $x=x_i$ с точностью до члена первого порядка относительно x

$$y(x) = y_i + (x-x_i)y_i' + (x-x_i)^2\frac{y''_i}{2} + (x-x_i)^3\frac{y'''_i}{3} + \ldots$$

Ошибка метода Эйлера пропорциональна h^2 (первому отброшенному члену ряда) и при удвоении числа n уменьшается в 4 раза.

Метод Эйлера модифицированный использует для вычисления интегралов $\int_{x_i}^{x_{i+1}} f(x,y)\,dx$, $i=0,1,\ldots,n-1$ классическую формулу прямоугольников:

$\int_{x_i}^{x_{i+1}} f(x,y)\,dx \approx h\cdot f\left[x_i + h/2, y_i\left(x_i + h/2\right)\right]$. Для определения значения y в т. $x_i + h/2$ используется классический метод Эйлера: $y\left(x_i + h/2\right) = y_i + f(x_i, y_i)\cdot h/2$. Тогда

$$y_{i+1} = y_i + h\cdot f\left[x_i + \frac{h}{2}, y_i + \frac{h}{2}\cdot f(x_i, y_i)\right], \quad i = 0,1,\ldots,n-1. \qquad (3.30)$$

Геометрическая интерпретация (рис. 3.14): Точка (x_{i+1},y_{i+1}) лежит на пересечении прямой $x = x_{i+1}$ и прямой, параллельной L и проходящей через исходную точку (x_i,y_i). Прямая L – касательная к интегральной кривой в точке $\left(x_i + \frac{h}{2}, y_i + \frac{h}{2}\cdot f(x_i, y_i)\right)$, образованной пересечением прямой $x = x_i + h/2$ и касательной к искомой кривой $y(x)$ в исходной точке (x_i,y_i).

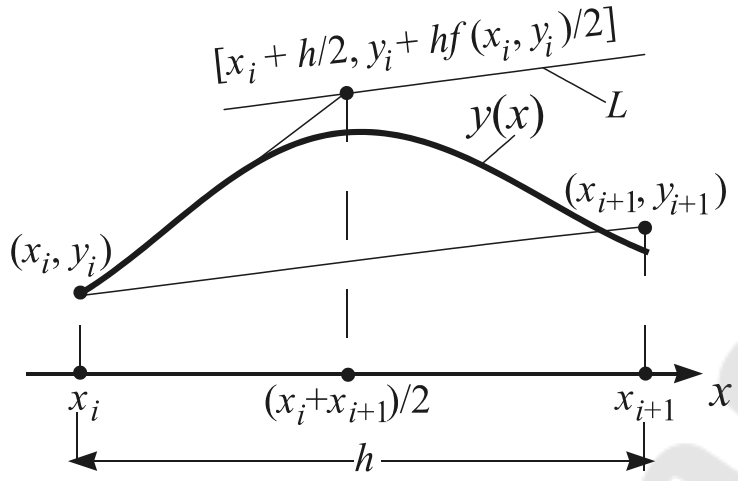


Рис. 3.14 Метод Эйлера модифицированный

Метод Эйлера исправленный. Здесь для вычисления интегралов $\int_{x_i}^{x_{i+1}} f(x,y)\,dx$, i=0,1,…,n-1 используется формула трапеций:

$\int_{x_i}^{x_{i+1}} f(x,y)\,dx \approx \frac{h}{2}\cdot[f(x_i,y_i)+f(x_i+h,y(x_i+h))]$, а значение $y(x_i+h)$ определяется с помощью классического метода Эйлера: $y(x_i+h)=y_i+h\cdot f(x_i,y_i)$. Тогда

$$y_{i+1}=y_i+\frac{h}{2}\cdot\left[f(x_i,y_i)+f(x_i+h,y_i+h\cdot f(x_i,y_i))\right],\ i=0,1,2,...,n-1. \quad (3.31)$$

Геометрическая интерпретация (рис.3.15): Точка (x_{i+1},y_{i+1}) – это точка пересечения прямой $x = x_{i+1}$ и прямой, параллельной L, которая проходит через исходную точку (x_i,y_i). Тангенс угла наклона прямой L к оси x равен среднему арифметическому тангенсов углов наклона касательных к кривой $y(x)$ в точках (x_i,y_i) и (x_i+h, y_i+h·f (x_i,y_i)).

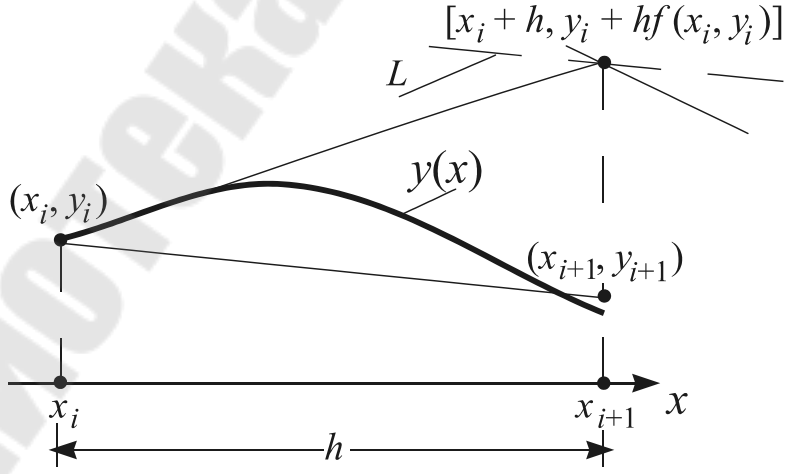


Рис. 3.15 Метод Эйлера исправленный

Модификации метода Эйлера значительно точнее классического, но требуют и большего объема вычислений: положение каждой точки искомой кривой определяется с помощью двукратного вычисления значения функции $f(x, y)$. Это методы второго порядка, т.к. их основные соотношения совпадают с разложением $y(x)$ в ряд Тейлора с точностью до члена второго порядка относительно x. Ошибка методов пропорциональна h^3 и при уменьшении величины шага вдвое уменьшается в 8 раз.

Метод Рунге-Кутта основан на использовании для вычисления интегралов $\int_{x_i}^{x_{i+1}} f(x,y)\,dx,\ i = 0,1,...,n-1$ формулы Симпсона:

$$\int_{x_i}^{x_{i+1}} f(x,y)dx = \frac{h}{6}\cdot\left\{ f(x_i,y_i) + 4\cdot f\left[x_i + \frac{h}{2}, y\left(x_i + \frac{h}{2}\right)\right] + f\left[x+h, y(x+h)\right]\right\}.$$

Определив значения $y(x_i + h/2),\ y(x_i + h)$ по методу Эйлера, получим

$$y_{i+1} = y_i + \frac{h}{6}\cdot\left\{ f(x_i,y_i) + 4\cdot f\left[x_i + \frac{h}{2}, y_i + \frac{h}{2}\cdot f(x_i,y_i)\right] + f\left[x_i + h, y + h\cdot f(x_i,y_i)\right]\right\},\ i$$

$=0,1,\ldots,n$.

Это формула метода Рунге-Кутта 3-го порядка. На практике чаще используется метод Рунге-Кутта 4-го порядка:

$$y_{i+1} = y_i + \frac{h}{6}\cdot(k_1 + 2k_2 + 2k_3 + k_4),\quad i = 0,1,2,...,n-1, \qquad (3.32)$$

где $k_1 = f(x_i, y_i);\quad k_2 = f\left(x_i + \frac{h}{2}, y_i + \frac{h}{2}\cdot k_1\right);\quad k_3 = f\left(x_i + \frac{h}{2}, y_i + \frac{h}{2}\cdot k_2\right);$

$k_4 = f(x_i + h, y_i + h\cdot k_3)$. Ошибка формулы (3.32) пропорциональна h^5.

Этот метод намного более точен, чем методы Эйлера, но требует и большего объема вычислений: положение точки $(x_{i+1},\ y_{i+1})$ определяется в результате 4-кратного вычисления значения функции $f(x,y)$. С появлением ЭВМ этот недостаток перестал быть существенным и метод Рунге-Кутта 4-го порядка применяется на практике чрезвычайно широко.

Все рассмотренные методы решения задачи Коши называются *одношаговыми*, т.к. каждая следующая точка (x_{i+1},y_{i+1}) искомой интегральной кривой определяется на основе информации только об одной предыдущей точке (x_i,y_i). Число микроотрезков $[x_i;\ x_{i+1}]$, на которые разбивается исходный отрезок $[x_0;x_n]$, определяется требуемой точностью вычислений. Для достижения нужной точности задача решается несколько раз при последовательно удваиваемом числе микроотрезков n. Точность считается достигнутой, если при начальном и удвоенном числе n значения y_i и y_{2i} (в совпадающих точках x)

отличаются не более чем на заданную величину: $\max\limits_{i=1,2,\ldots,n}\left|y_i^{(n)}-y_{2i}^{(2n)}\right|<\varepsilon$.

Методы прогноза и коррекции - это общее название многошаговых методов численного решения задачи Коши. Они используют для определения положения точек (x_{i+1},y_{i+1}) искомой интегральной кривой информацию о положении *нескольких* предыдущих точек. Положение точки (x_{i+1},y_{i+1}) вначале прогнозируется с учетом известного положения двух или более предыдущих точек, а затем корректируется с помощью итерационной процедуры. Наиболее часто применяемый на практике метод этого семейства использует для прогноза значения y_{i+1} формулу:

$$y_{i+1}^{(0)}=y_{i-1}+2\cdot h\cdot f(x_i,y_i),\quad i=1,2,\ldots,n-1, \tag{3.33}$$

а для коррекции применяется итерационный процесс

$$y_{i+1}^{(k)}=y_i+\frac{h}{2}\cdot\left[f(x_i,y_i)+f(x_{i+1},y_{i+1}^{(k-1)})\right],\ i=1,2,\ldots,n-1;\ k=1,2,\ldots \tag{3.34}$$

Процесс (3.34) прекращается в момент выполнения неравенства $\left|y_{i+1}^{(k)}-y_{i+1}^{(k-1)}\right|<\varepsilon$. Методом (3.33)-(3.34) невозможно начать решение задачи, т.к. в распоряжении имеется единственная точка (x_0,y_0) искомой интегральной кривой, а формула (3.33) ориентирована на две известные точки. Поэтому первый шаг (определение y_1) делается обычно по методу Эйлера или Рунге-Кутта. Метод прогноза и коррекции позволяет оценить приемлемость выбранного шага интегрирования в ходе расчетов: если процесс (3.34) сходится за 2-3 итерации, то считается, что шаг выбран верно; если необходима всего одна итерация, то шаг следует увеличить; если требуется больше трех итераций, то шаг необходимо уменьшить. При изменении шага формулу прогноза (3.33) применить не удается, приходится делать один или несколько шагов с помощью одношагового метода.

Алгоритм метода (3.33)-(3.34) должен включать: алгоритм одного из одношаговых методов, алгоритм вычисления прогнозируемых значений y_i по формуле (3.33) и алгоритм реализации итерационного процесса (3.34). Несмотря на сложность, этот алгоритм иногда требует меньших затрат машинного времени, чем алгоритмы одношаговых методов за счет меньшего количества вычислений значений $f(x,y)$, варьирования шага, отсутствия необходимости повторных решений задачи.

3.6.2 Решение систем обыкновенных дифференциальных уравнений

Для решения систем обыкновенных дифференциальных уравнений используются те же методы, что и для решения одного уравнения: методы Эйлера, Рунге-Кутта, прогноза и коррекции. В качестве примера рассмотрим задачу решения системы двух обыкновенных дифференциальных уравнений:

$$\begin{cases} y_1' = f_1(x, y_1, y_2); \\ y_2' = f_2(x, y_1, y_2); \end{cases} \quad (3.35)$$

Задача заключается в нахождении интегральных кривых $y_1(x)$ и $y_2(x)$, удовлетворяющих начальным условиям:

$$y_1(x_0) = y_{10} \ , \quad y_2(x_0) = y_{20} \ . \quad (3.36)$$

Задача (1.35)-(1.36) также называется задачей Коши и на практике обычно решается для фиксированного отрезка $[x_0;x_n]$ оси x, который в ходе решения разбивается на микроотрезки $[x_i;\ x_{i+1}]$, $i=1,2,....,n$ длиной $h=(x_n-x_0)/n$.

Число микроотрезков n определяется заданной точностью решения.

Классический метод Эйлера предусматривает определение положения следующих точек искомых интегральных кривых $y_1(x)$ и $y_2(x)$ как точек пересечения прямой $x = x_i + h$ с касательными к соответствующим кривым в точках (x_i, y_{1i}) и (x_i, y_{2i}), $i = 0,1,...,n-1$, т.е.

$$\begin{cases} y_{1,i+1} = y_{1i} + h \cdot f_1(x_i, y_{1i}, y_{2i}); \\ y_{2,i+1} = y_{2i} + h \cdot f_2(x_i, y_{1i}, y_{2i}); \end{cases}, \quad i = 0,1,..,n-1 \quad (3.37)$$

Гораздо чаще классического на практике применяются модификации метода Эйлера: *модифицированный*:

$$\begin{cases} y_{1,i+1} = y_{1i} + h \cdot k_{12}; \\ y_{2,i+1} = y_{2i} + h \cdot k_{22}; \end{cases}, \quad i=0,1,\ldots,n-1, \quad (3.38)$$

где $k_{12} = f_1\left(x_i + \frac{h}{2}, y_{1i} + \frac{h}{2} \cdot k_{11}, y_{2i} + \frac{h}{2} \cdot k_{21}\right)$; $k_{11} = f_1(x_i, y_{1i}, y_{2i})$;

$k_{21} = f_2(x_i, y_{1i}, y_{2i})$; $k_{22} = f_2\left(x_i + \frac{h}{2}, y_{1i} + \frac{h}{2} \cdot k_{11}, y_{2i} + \frac{h}{2} \cdot k_{21}\right)$ и *исправленный*:

$$\begin{cases} y_{1,i+1} = y_{1i} + \dfrac{h}{2}\cdot(k_{11}+k_{12}); \\ y_{2,i+1} = y_{2i} + \dfrac{h}{2}\cdot(k_{21}+k_{22}); \end{cases}, \quad i=0,1,\ldots,n\text{-}1, \qquad (3.39)$$

где $k_{11}=f_1(x_i,y_{1i},y_{2i})$; $k_{21}=f_2(x_i,y_{1i},y_{2i})$; $k_{12}=f_1(x_i+h,y_{1i}+h\cdot k_{11},y_{2i}+h\cdot k_{21})$; $k_{22}=f_2(x_i+h,y_{1i}+h\cdot k_{11},y_{2i}+h\cdot k_{21})$.

Метод Рунге-Кутта применяется на практике еще более широко, чем модификации метода Эйлера. Его основные соотношения для решения задачи (3.35)-(3.36) можно получить из соотношения (3.32) аналогично тому, как получены выражения (3.37)-(3.39) из (3.29)-(3.31) соответственно:

$$\begin{cases} y_{1,i+1} = y_{1i} + \dfrac{h}{6}\cdot(k_{11}+2k_{12}+2k_{13}+k_{14}); \\ y_{2,i+1} = y_{2i} + \dfrac{h}{6}\cdot(k_{21}+2k_{22}+2k_{23}+k_{24}); \end{cases}, \quad i=0,1,\ldots,n\text{-}1, \qquad (3.40)$$

где $k_{11}=f_1(x_i,y_{1i},y_{2i})$; $k_{21}=f_2(x_i,y_{1i},y_{2i})$,

$$k_{12}=f_1\left(x_i+\frac{h}{2},y_{1i}+\frac{h}{2}\cdot k_{11},y_{2i}+\frac{h}{2}\cdot k_{21}\right);\ k_{22}=f_2\left(x_i+\frac{h}{2},y_{1i}+\frac{h}{2}\cdot k_{11},y_{2i}+\frac{h}{2}\cdot k_{21}\right);$$

$$k_{13}=f_1\left(x_i+\frac{h}{2},y_{1i}+\frac{h}{2}\cdot k_{12},y_{2i}+\frac{h}{2}\cdot k_{22}\right);\ k_{23}=f_2\left(x_i+\frac{h}{2},y_{1i}+\frac{h}{2}\cdot k_{12},y_{2i}+\frac{h}{2}\cdot k_{22}\right);$$

$$k_{14}=f_1(x_i+h,y_{1i}+h\cdot k_{13},y_{2i}+h\cdot k_{23});\ k_{24}=f_2(x_i+h,y_{1i}+h\cdot k_{13},y_{2i}+h\cdot k_{23}).$$

Заданная точность аппроксимации искомых интегральных кривых $y_1(x)$ и $y_2(x)$ ломаными Эйлера или кривыми, состоящими из отрезков квадратичных парабол, достигается в результате использования приема последовательного удвоения числа и элементарных интервалов интегрирования. Точность считается достигнутой, если при начальном и удвоенном n значения y_{1i}, y_{2i} в совпадающих точках x отличаются друг от друга не более чем на заданную величину: $\max\limits_{i=1,2,\ldots,n}\left\{\left|y_{1_i}^{(n)}-y_{1_{2i}}^{(2n)}\right|,\left|y_{2_i}^{(n)}-y_{2_{2i}}^{(2n)}\right|\right\}<\varepsilon$. При неудачном выборе начального значения n достижения нужной точности может потребовать значительных затрат времени.

Метод прогноза и коррекции также может быть использован для решения задачи (3.35)-(3.36). Формула прогноза значений $y_{1,i+1}$, $y_{2,i+1}$ (см. (3.33)):

$$\begin{cases} y_{1,i+1}^{(0)} = y_{1,i-1} + 2h\cdot f_1(x_i,y_{1i},y_{2i}) \\ y_{2,i+1}^{(0)} = y_{2,i-1} + 2h\cdot f_2(x_i,y_{1i},y_{2i}) \end{cases}, \quad i=1,2,\ldots,n\text{-}1. \qquad (3.41)$$

Формула коррекции спрогнозированных значений $y_{1,i+1}, y_{2,i+1}$ (см.

(3.34)):

$$\begin{cases} y_{1,i+1}^{(k)} = y_{1i} + \dfrac{h}{2}\cdot\left[f_1(x_i, y_{1i}, y_{2i}) + f_1(x_i + h, y_{1,i+1}^{(k-1)}, y_{2,i+1}^{(k-1)})\right] \\ y_{2,i+1}^{(k)} = y_{2i} + \dfrac{h}{2}\cdot\left[f_2(x_i, y_{1i}, y_{2i}) + f_2(x_i + h, y_{1,i+1}^{(k-1)}, y_{2,i+1}^{(k-1)})\right] \end{cases}, \quad i=0,\ldots,n\text{-}1;\ k=1,\ldots \quad (3.42)$$

до момента выполнения неравенства:

$$\max\left\{\left|y_{1,i+1}^{(k)} - y_{1,i+1}^{(k-1)}\right|, \left|y_{2,i+1}^{(k)} - y_{2,i+1}^{(k-1)}\right|\right\} < \varepsilon. \quad (3.43)$$

Алгоритм этого метода предусматривает использование одношагового метода для осуществления первого шага (определения y_{11}, y_{21}) и первого шага после изменения значения h, когда условие (3.43) выполняется уже при $k=1$ (шаг h можно увеличить) или при $k>3$ (значение h следует уменьшить).

Рассмотренные методы решения систем обыкновенных дифференциальных уравнений могут быть использованы и для решения уравнений высоких порядков. Например, задача решения уравнения $y'' = f(x, y', y)$ при условиях $y(x_0) = y_0;\ y'(x_0) = y_0';$ заменой переменных $z = y' \to y'(x_0) = y_0' = z_0$ сводится к задаче решения системы двух уравнений первого порядка:

$$\begin{cases} z' = f(x, z, y); \\ y' = z; \end{cases} \text{ при условиях } \begin{array}{l} y(x_0) = y_0; \\ z(x_0) = z_0. \end{array}$$

3.7 Решение уравнений в частных производных

Дифференциальные уравнения в частных производных применяются при математическом описании статики и динамики объектов с распределенными параметрами. Они используются для описания процессов в гидродинамике, теплопередаче, диффузионной и химической кинетике, ядерной физике, аэродинамике.

Рассмотрим одну из наиболее простых задач решения дифференциальных уравнений в частных производных - задачу решения линейного уравнения 2-го порядка с двумя независимыми переменными: найти решение $U=U(x,y)$ уравнения

$$A(x,y)\cdot\frac{\partial^2 U}{\partial x^2} + B(x,y)\cdot\frac{\partial^2 U}{\partial x\,\partial y} + C(x,y)\cdot\frac{\partial^2 U}{\partial y^2} + D(x,y)\cdot\frac{\partial U}{\partial x} + E(x,y)\cdot\frac{\partial U}{\partial y} + F(x,y)\cdot U = G(x,y), \quad (3.44)$$

удовлетворяющее условиям:

$$U(x, y_0) = \varphi(x); \quad \frac{\partial U(x, y_0)}{\partial y} = \psi(x). \qquad (3.45)$$

Здесь $A(x,y)$, $B(x,y)$, $C(x,y)$, $D(x,y)$, $E(x,y)$, $F(x,y)$, $G(x,y)$ - заданные функции.

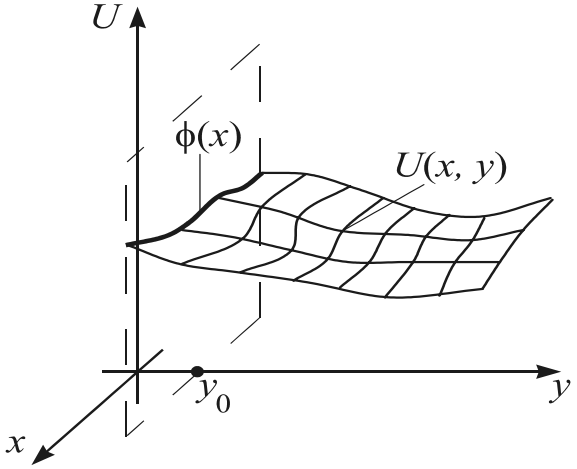


Рис. 3.16 Решение задачи (3.44)–(3.45)

Решение уравнения (3.44) - это поверхность $U(x,y)$ (рис. 3.16), линия пересечения которой с плоскостью $y=y_0$ соответствует функции $\varphi(x)$, а скорость изменения вдоль оси y при $y=y_0$ - функции $\psi(x)$.

Условия (3.45) называются начальными условиями, а задача (3.44) - (3.45) - задачей Коши. Часто при записи подобных задач производные изображаются упрощенно: U_{xx}, U_{xy}, U_{yy}, U_x, U_y. На практике такие задачи как правило решаются численно с помощью разновидностей метода конечных разностей (жаргонное название – метод сеток).

Условия (3.45) и им подобные определяют некоторую область плоскости xy (рис.3.17), всем точкам которой требуется поставить в соответствие значения $U(x,y)$, удовлетворяющие уравнению (3.44).

Предположим, что граница этой области R образована прямыми, параллельными осям x и y. Сетка формируется в результате разбиения отрезка H оси x на n микроотрезков длиной $h=H/n$, а отрезка L оси y - на m микроотрезков длиной $l=L/m$.

В области образуется $(n-1)\cdot(m-1)$ пересечений границ микроотрезков, называемых узлами сетки. В каждом из них необходимо определить значение функции $U(x_i,y_i) = U(i\cdot h, j\cdot l) = U_{ij}$, $i=1,2,\ldots,n-1$; $j=1,2,\ldots,m-1$.

При достаточно больших значениях n и m значения U_{ij} позволяют судить о характере искомой поверхности $U(x,y)$. Значения $U(x,y)$ на границе области: U_{0j}, U_{i0}, U_{nj}, U_{im}, $i=1,\ldots,n-1$; $j=1,\ldots,m-1$, - определяются из условий (3.45), а внутри области – путем замены производных в уравнении (3.44) конечно-раз-ностными отношениями:

правая производная $U_x(x_i, y_j) \approx \frac{U(x_i + h, y_j) - U(x_i, y_j)}{h} = \frac{U_{(i+1),j} - U_{ij}}{h};$ (3.46)

левая производная $U_x(x_i,y_j) \approx \frac{U(x_i,y_j)-U(x_i-h,y_j)}{h} = \frac{U_{ij}-U_{(i-1),j}}{h}$; (3.47)

$$U_{xx}(x_i,y_j) \approx \frac{U_x(x_i+h,y_j)-U_x(x_i,y_j)}{h} \approx \frac{(U_{(i+1),j}-U_{ij})/h-(U_{ij}-U_{(i-1),j})/h}{h} = \frac{U_{(i-1),j}-2U_{ij}+U_{(i+1),j}}{h^2}. \quad (3.48)$$

Формула (3.48), полученная путем вычисления первых производных как левых, а второй - как правой (или наоборот), симметрична относительно т.(x_i,y_j) и называется центрально-разностным отношением. Именно такие выражения обычно используются для аппроксимации вторых производных. Производные U_{yy}, U_y в уравнении (3.44) заменяются выражениями:

$$U_y(x_i,y_i) \approx \frac{U_{i,(j+1)}-U_{ij}}{l} \approx \frac{U_{ij}-U_{i,(j-1)}}{l}, \quad (3.49)$$

$$U_{yy}(x_i,y_i) \approx \frac{U_{i,(j-1)}-2U_{ij}+U_{i,(j+1)}}{l^2}. \quad (3.50)$$

При вычислении приближенных значений производной U_{xy} используются формулы (3.46), (3.47), (3.49), причем если по x берется правая производная, то по y - левая и наоборот.

Такая замена производных функций $U(x,y)$ в уравнении (3.44) и условиях (3.45) для каждого узла (i,j) сформированной сетки приведет к замене задачи (3.44)-(3.45) задачей решения системы $(n+1)\cdot(m+1)$ линейных уравнений с $(n+1)\cdot(m+1)$ неизвестными $U_{ij}=U(x_i,y_j)$, $i=0,1,...,n$, $j=0,1...,m$, которую чаще всего решают методом Зейделя. Заданная точность решения достигается в результате использования приема последовательного удвоения предварительно выбранных значений n и m.

В практических задачах граница области, определяемой условиями (3.45) редко бывает прямоугольной и обычно аппроксимируется ломаной линией, непрерывные фрагменты которой параллельны осям x и y. В зависимости от значений A, B и C уравнения вида (3.45) подразделяют на эллиптические ($B^2 - 4\cdot A\cdot C < 0$), параболические ($B^2 - 4\cdot A\cdot C = 0$) и гиперболические ($B^2 - 4\cdot A\cdot C > 0$).

Рассмотрим метод конечных разностей более подробно на примере решения параболического уравнения - уравнения теплопроводности $\frac{\partial T(x,\tau)}{\partial \tau} = a\cdot\frac{\partial^2 T(x,\tau)}{\partial x^2}$, характеризующего процесс распространения тепла в бесконечной пластине толщиной R, расположенной перпендикулярно оси x.

В этом уравнении $a = \frac{\lambda}{c \cdot \rho}$ – температуропроводность материала пластины.

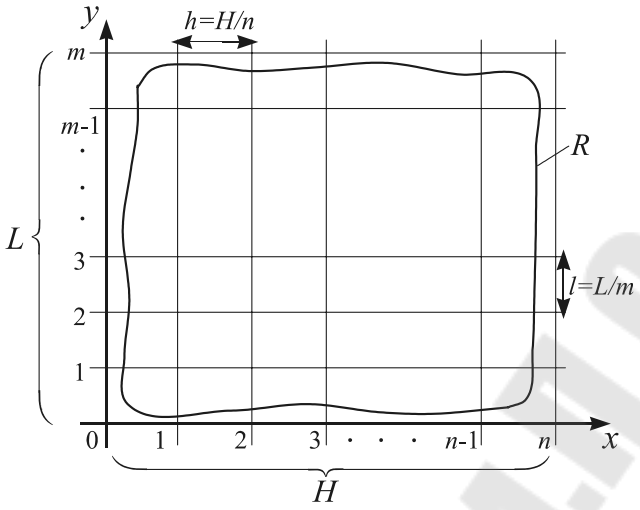


Рис. 3.17 Иллюстрация к методу сеток

Предполагая, что на поверхности пластины, т.е. при $x=0$ и $x=R$ поддерживается температура $T_{\text{п}}$, а распределение температуры внутри пластины в начальный момент времени $\tau = 0$ характеризуется функцией $f(x)$, и что $a = \text{const} > 0$, получим задачу: решить уравнение $T'_\tau = a \cdot T''_{xx}$ при условиях: $T(0,\tau)=T(R,\tau)=T_{\text{п}}$ (граничные); $T(x,0)=f(x)$, $0 \le x \le R$ (начальное).

Для решения этой задачи методом конечных разностей разобьём отрезок $[0;R]$ оси x на n микроотрезков длиной $h=R/n$. Значение τ в постановке задачи не ограничено сверху, поэтому вдоль оси τ, начиная с $\tau=0$, будем последовательно откладывать микроотрезки произвольно выбранной длины l. При переходе к разностной форме записи граничные условия примут вид: $T_{0j}=T_{nj}=T_{\text{п}}$; $j = 1,2,3, \ldots$, а начальное условие: $T_{i0}=f(ih)$, $i = 1,2, \ldots, (n-1)$. Уравнение перепишется в виде:

$$\frac{T_{i,(j+1)} - T_{ij}}{l} = a \cdot \frac{T_{(i-1),j} - 2T_{ij} + T_{(i+1),j}}{h^2} .$$

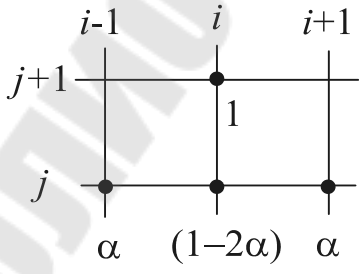


Рис. 3.18 Трафарет уравнения (3.51)

Обозначив $\alpha = a \cdot l/h^2$, после преобразований получим:

$$T_{i,(j+1)} = \alpha \cdot T_{(i-1),j} + (1 - 2 \cdot \alpha) \cdot T_{ij} + \alpha \cdot T_{(i+1),j},$$

$i=1,2,\ldots,n-1$, $j=0,1,2,\ldots$ (3.51)

Способ аппроксимации дифференциального уравнения разностным принято иллюстрировать так называемым трафаретом, где

показано, сколько узлов сетки включает уравнение и каковы значения коэффициентов при соответствующих T_{ij} (см. рис. 3.18, 3.19).

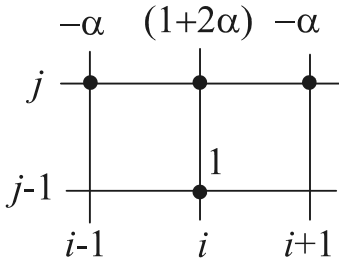


Рис. 3.19 Трафарет уравнения (3.52)

При выбранном способе преобразования задачи к разностной форме она решается по так называемой явной схеме: значения $T(x,\tau)$ в узлах нижней строки сформированной сетки (значения T_{i0}, i=1, 2, ..., (n-1)) заданы начальным условием, а в узлах второй и последующих строк - определяются непосредственно из уравнения (3.51) через значения $T(x,\tau)$ в 3-х ближайших узлах предыдущей строки, например при $j = 0$: $T_{i,1} = \alpha \cdot T_{(i-1),0} + (1-2\cdot\alpha)\cdot T_{i0} + \alpha \cdot T_{(i+1),0}$, i=1,…,(n-1). Если же при переходе от дифференциального уравнения к разностному определить T'_τ не через правую, а через левую конечную разность, то вместо (3.51) получим уравнение:

$$T_{i,(j-1)} = -\alpha \cdot T_{(i-1),j} + (1+2\cdot\alpha)\cdot T_{ij} - \alpha \cdot T_{(i+1),j},\ i=1,...,\ n\text{-}1,\ j=1,..., \qquad (3.52)$$

которое невозможно решить по явной схеме, т.к. в его трафарете известно положение единственного узла ($T_{i,(j-1)}$). В этом случае применяется неявная схема решения задачи: для первой строки по оси τ (j=1), используя граничные и начальные условия, получим:

$$\begin{array}{ll}
T_{10} = -\alpha\cdot T_{01} + (1+2\cdot\alpha)\cdot T_{11} - \alpha\cdot T_{21} & \rightarrow \\
T_{20} = -\alpha\cdot T_{11} + (1+2\cdot\alpha)\cdot T_{21} - \alpha\cdot T_{31} & \rightarrow \\
T_{30} = -\alpha\cdot T_{21} + (1+2\cdot\alpha)\cdot T_{31} - \alpha\cdot T_{41} & \rightarrow \\
\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots & \rightarrow \\
T_{(n-1),0} = -\alpha\cdot T_{(n-2),1} + (1+2\cdot\alpha)\cdot T_{(n-1),1} - \alpha\cdot T_{n1} & \rightarrow
\end{array}
\left\{
\begin{array}{l}
(1+2\cdot\alpha)\cdot T_{11} - \alpha\cdot T_{21} = f(h) + \alpha\cdot T_{\text{п}} \\
-\alpha\cdot T_{11} + (1+2\cdot\alpha)\cdot T_{21} - \alpha\cdot T_{31} = f(2h) \\
-\alpha\cdot T_{21} + (1+2\cdot\alpha)\cdot T_{31} - \alpha\cdot T_{41} = f(3h) \\
\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\
-\alpha\cdot T_{(n-2),1} + (1+2\cdot\alpha)\cdot T_{(n-1),1} = f(nh-h) + \alpha\cdot T_{\text{п}}
\end{array}
\right. , \qquad (3.53)$$

т.е. систему (n-1) линейных уравнений с (n-1) неизвестными T_{11}, T_{21},..., $T_{(n\text{-}1),1}$. Диагональные элементы матрицы этой системы равны (1+2·α), то есть превосходят по абсолютной величине сумму абсолютных величин всех других элементов соответствующей строки матрицы системы (α или 2·α), следовательно эту систему можно без преобразований решать методом Зейделя.

Таким же образом можно найти значения T_{ij} в узлах второй и последующих строк сетки по оси τ, то есть решение задачи по неявной схеме приводит к необходимости последовательно формировать и решать линейные системы вида (3.53). Для решения подобных систем разработана специальная разновидность метода Гаусса – *метод прогонки*.

Метод прогонки предназначен для решения систем линейных

уравнений с трехдиагональными матрицами:

$$\alpha_i \cdot x_{i-1} - \delta_i \cdot x_i + \beta_i \cdot x_{i+1} = -\varphi_i, \quad i = 1,2,..., n-1,$$

причем значения x_0 и x_n определяются из рекуррентных соотношений, которые при решении дифференциальных уравнений по неявной схеме образуются из граничных условий: $x_0 = \theta_1 \cdot x_1 + \mu_1$, $x_n = \theta_2 \cdot x_{n-1} + \mu_2$. Решение такой системы имеет вид: $x_i = a_{i+1} \cdot x_{i+1} + b_{i+1}$, $i = n-1,...,1$, где коэффициенты a_{i+1} и b_{i+1} определяются по формулам $a_{i+1} = \beta_i / (\delta_i - \alpha_i \cdot a_i)$, $b_{i+1} = (\alpha_i \cdot b_i + \varphi_i) / (\delta_i - \alpha_i \cdot a_i)$, $i = 1,..., n-1$, причем $a_1 = \theta_1$, $b_1 = \mu_1$.

Решение системы (3.53) методом прогонки имеет вид:

$T_{ij} = a_{i+1} \cdot T_{i+1,j} + b_{i+1}$, $i = n-1,...,1$, где $a_{i+1} = \alpha / [(1+2 \cdot \alpha) - a_i \cdot \alpha]$,

$b_{i+1} = (\alpha \cdot b_i + T_{i,j-1}) / [(1+2 \cdot \alpha) - a_i \cdot \alpha]$, $i = 1,..., n-1$, $a_1 = 0$, $b_1 = T_{0,j}$.

На первый взгляд кажется, что явная схема предпочтительнее неявной, однако решение, получаемое с использованием уравнения (3.52) устойчиво и сходится при $h \to 0$, $l \to 0$ к решению исходной задачи при любом соотношении значений шагов по осям x и τ, а получаемое с использованием уравнения (3.51) - лишь при $l < h^2/2$, что не всегда удобно (при неявной схеме можно делать большие шаги по времени, а при явной приходится делать чрезвычайно мелкие).

Если, согласно трафарету, в разностном уравнении один неизвестный узел, оно решается по явной схеме, если более одного – по неявной.

4 МЕТОДЫ ОПТИМИЗАЦИИ

Оптимизация в широком смысле слова - это поиск лучшего из возможных вариантов. Применительно к емкостному реактору периодического действия, в котором реализуется эндотермическая химическая реакция (рис.4.1), это может быть поиск наилучшей конструкции аппарата (диаметр, высота, объем, поверхность теплообмена) или наиболее приемлемых параметров ведения технологического процесса (температура t^o, продолжительность τ, начальные концентрации компонентов c_A, c_B). Поиск всегда предполагает наличие цели, например, максимум выхода продукта, минимальное время достижения заданной концентрации продукта, минимальный расход компонента *A* при заданном выходе продукта. *Методы оптимизации* – это численные методы решения задач оптимизации – задач, имеющих множество допустимых решений, из которых необходимо выбрать одно, лучшее в каком-либо смысле.

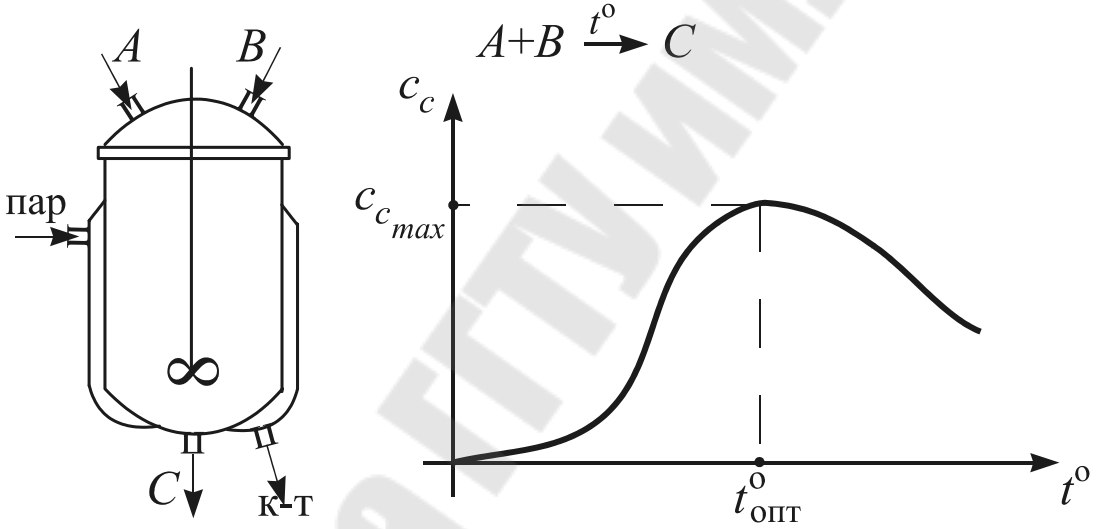


Рис. 4.1 Пример объекта оптимизации

4.1 Структура и постановка задач оптимизации

Формулировка задачи оптимизации включает три этапа: 1) словесное представление о параметрах задачи, множестве ее решений и поставленной цели; 2) запись критерия оптимальности (целевой функции) как функции параметров задачи; 3) запись условий, определяющих область допустимых значений параметров.

Параметры задачи x_i, i=1,2,…,n - это переменные, значения которых необходимо определить в результате ее решения (на рис. 4.1 – температура в реакторе). Критерий оптимальности может быть

представлен в виде функции параметров (целевой функции) $f(\bar{X})$, где $\bar{X}=(x_1,x_2,...,x_n)$ - одно из допустимых решений задачи, или функционала $I(\bar{X},t)$, который при фиксированных значениях параметров задачи представляет собой не число, а функцию времени или пространственной координаты, например,

$$f(\bar{X})=\sum_{i=1}^{n}(x_i-x_i^{(0)})^2, \qquad I(\bar{X},t)=\int_0^t \varphi(\bar{X},t)dt\,.$$

Область допустимых значений параметров $x_i,\ i=1,2,...,n$ может быть определена с помощью ограничений $g_j(\bar{X})\ge 0;\ j=1,2,...,m$ (условия типа неравенств) и связей $h_k(\bar{X})=0;\ k=1,2,...,p$ (условия типа равенств).

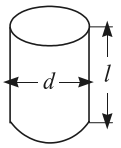
Таким образом, обобщенная постановка оптимизационной задачи имеет вид: найти решение $\bar{X}^*=(x_1^*,x_2^*,...,x_n^*)$, которому соответствует экстремум

функции $f(\bar{X})$ (минимум или максимум), при выполнении условий:

$$g_j(\bar{X})\ge 0;\ j=1,2,...,m; \qquad (4.1)$$

$$h_k(\bar{X})=0;\ k=1,2,...,p. \qquad (4.2)$$

Пример. Задача проектирования цилиндрической емкости.



Словесная постановка задачи: найти такие значения диаметра и высоты цилиндра, чтобы при фиксированном объеме V общая длина сварных швов была минимальна.

Критерий оптимальности: сварные швы - это две окружности и высота цилиндра, поэтому $f(d,l)=2\pi d+l$.

Множество допустимых решений задачи: диаметр и высота емкости не могут быть отрицательными числами, т.е. $d>0$, $l>0$; объем проектируемой емкости фиксирован и равен V, т.е. $\pi d^2 l/4=V$.

Таким образом, задача состоит в определении таких значений d^* и l^*, что функция $f(d,l)=2\pi d+l$ достигает минимума и выполняются условия: $\pi d^2 l/4=V$; $d>0$; $l>0$.

Задачи оптимизации классифицируются по следующим признакам:

1) наличие или отсутствие ограничений и связей – задачи *на безусловный экстремум* (условия (4.1), (4.2) отсутствуют) и *на условный экстремум*

(имеется условие (4.1), условие (4.2) или оба);

2) вид критерия оптимальности – *вариационные* задачи (критерий - функционал) и задачи *математического программирования*

(критерий - функция);

3) характер функций *f*, *g*, *h* – задачи *линейного программирования* (все функции - линейные) и задачи *нелинейного программирования* (хотя бы одна из функций - нелинейная).

4) характер параметров – если параметры задачи могут принимать только строго определенные значения то это задача *дискретного (целочисленного) программирования*, а если число этих значений конечно – *комбинаторная* задача.

4.2 Основные типы вычислительных процедур оптимизации

Теоретической основой методов решения задач оптимизации являются условия оптимальности решения различных типов задач, т.е. условия, при которых критерий оптимальности той или иной задачи достигает минимального или максимального значения. Условия оптимальности подразделяются на необходимые и достаточные.

Примеры необходимых условий оптимальности (условие *A* необходимо для выполнения *B*, если при выполнении *B* всегда выполняется *A*):

- если дифференцируемая функция $f(x)$ имеет в точке $x=x^*$ экстремум, то $f'(x^*)\equiv 0$;
- для дифференцируемой функции многих переменных $f(x_1,x_2,...,x_n)$ необходимое условие экстремума $\frac{\partial f(x_1,x_2,...,x_n)}{\partial x_i}=0,\ i=1,2,...,n$;
- если в окрестности некоторой точки x^* ($x\in[x^*-\delta, x^*+\delta]$, $\delta>0$) $f(x)>f(x^*)$, то в точке $x=x^*$ функция $f(x)$ имеет локальный минимум (в окрестности другой точки x значения $f(x)$ могут быть еще меньше).

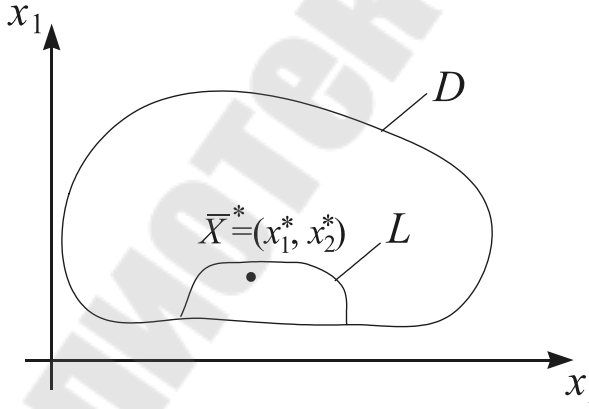


Рис. 4.2 Необходимое условие оптимальности

Обобщенное необходимое условие оптимальности (рис. 4.2): для того, чтобы $\bar{x}^*$ был лучшим элементом множества *D* решений некоторой оптимизационной задачи необходимо, чтобы он был лучшим элементом подмножества *L*, которое образует_окрестность точки $\bar{x}^*$ (необходимым условием

существования глобального экстремума целевой функции задачи является существование хотя бы одного локального экстремума).

Примеры достаточных условий оптимальности (условие A достаточно для выполнения B если при выполнении A всегда выполняется B):

- если в точке $x=x^*$ функция $f(x)$ имеет минимум, то в некоторой окрестности этой точки $x\in [x^*-\delta, x^*+\delta]$, $\delta>0$ выполняется условие $f(x) > f(x^*)$;

- если $f'(x^*)=0$ и $f''(x^*)\neq 0$, то в т. x^* $f(x)$ имеет экстремум (при $f''(x^*)>0$ - минимум, при $f''(x^*)<0$ - максимум);

- для того, чтобы дифференцируемая функция $f(x_1,x_2,...,x_n)$ имела в точ-ке $(x_1^*, x_2^*,..., x_n^*)$ минимум достаточно, чтобы $\frac{\partial f(x_1^*, x_2^*,..., x_n^*)}{\partial x_i}=0,\ i=1,2,...,n$ и все миноры матрицы Гессе

$$\begin{pmatrix} \frac{\partial^2 f(x_1^*,...,x_n^*)}{\partial x_1^2} & \frac{\partial^2 f(x_1^*,...,x_n^*)}{\partial x_1 \partial x_2} & ..... & \frac{\partial^2 f(x_1^*,...,x_n^*)}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f(x_1^*,...,x_n^*)}{\partial x_2 \partial x_1} & \frac{\partial^2 f(x_1^*,...,x_n^*)}{\partial x_2^2} & ..... & \frac{\partial^2 f(x_1^*,...,x_n^*)}{\partial x_2 \partial x_n} \\ ..... & ..... & ..... & ..... \\ \frac{\partial^2 f(x_1^*,...,x_n^*)}{\partial x_n \partial x_1} & \frac{\partial^2 f(x_1^*,...,x_n^*)}{\partial x_n \partial x_2} & ..... & \frac{\partial^2 f(x_1^*,...,x_n^*)}{\partial x_n^2} \end{pmatrix}$$

были положительны (если миноры нечетного порядка отрицательны, а четного - положительны, то в т.$(x_1^*,x_2^*,...,x_n^*)$ $f(x_1,x_2,...,x_n)$ имеет максимум).

Обобщенное достаточное условие оптимальности (рис. 4.3): для того, чтобы $\bar{X}^*$ был лучшим элементом множества D допустимых решений рассматриваемой задачи, достаточно чтобы $\bar{X}^*$ был лучшим элементом множества $V\supset D$ (если $\bar{X}^*\in D$ - глобальный экстремум функции $f(x_1,x_2)$ на множестве V, то $\bar{X}^*$ является глобальным экстремумом $f(x_1,x_2)$ и на множестве D).

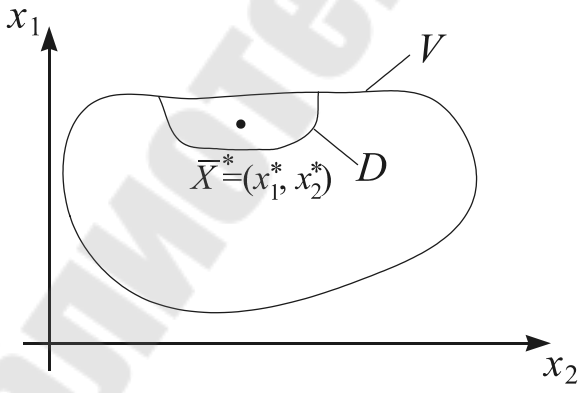


Рис. 4.3 Достаточное условие оптимальности

После того как сформулированы условия оптимальности критерия конкретной задачи, необходимо организовать

вычислительную процедуру поиска оптимального решения. Имеется пять основных типов вычислительных процедур решения задач оптимизации, на основе которых разрабатываются методы оптимизации:

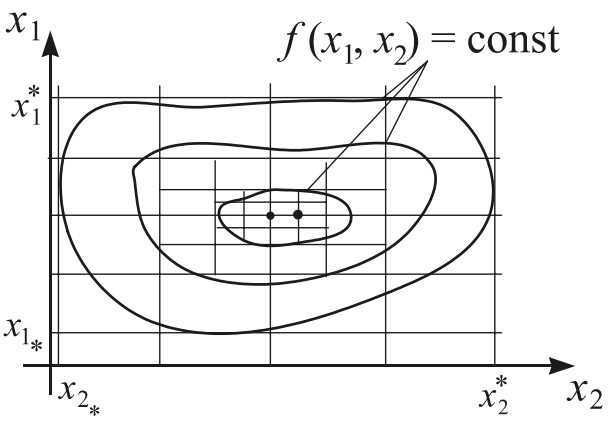


Рис. 4.4 Сетка значений аргументов $f(x_1,x_2)$

1. *Сравнение значений целевой функции на сетке значений аргументов.* Сетка образуется в результате разбиения областей допустимых значений аргументов на равные интервалы (рис. 4.4). Оптимальному решению соответствует минимальное или максимальное значение целевой функции в “узлах” сетки. Процедура обычно применяется многократно: вначале шаг сетки “крупный”, а затем вокруг лучшей точки строится более “мелкая” сетка.

Аналогом этой процедуры для задач дискретного программирования и комбинаторных является поиск лучшего допустимого решения путем полного или локального перебора.

2. *Использование необходимых условий экстремума* целевой функции, т.е. формирование и решение систем уравнений вида

$$\frac{\partial f(x_1, x_2, ..., x_n)}{\partial x_i} = 0, \quad i = 1,2,..., n .$$

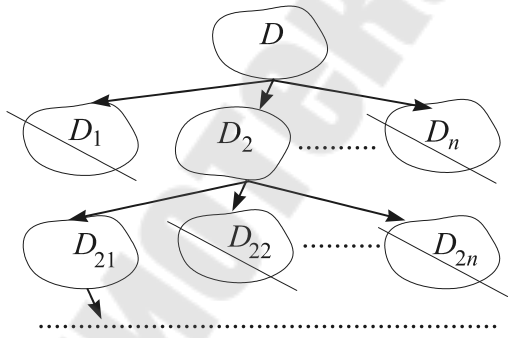


Рис. 4.5 Отсечение множеств неоптимальных решений

3. *Использование достаточных условий оптимальности*: образуется вспомогательная задача, множество решений которой шире допустимого, а критерий оптимальности на допустимом множестве совпадает с критерием исходной задачи (например, задача условной оптимизации заменяется задачей безусловной оптимизации, в

критерий которой вводится “штраф” за выход из допустимой области, т.е. за невыполнение условий (4.1), (4.2)).

4. *Отсечение множеств заведомо неоптимальных решений* (рис. 4.5) на основе правил, различных для каждой конкретной задачи (метод "золотого сечения", ветвей и границ для экстремальных комбинаторных задач).

5. *Построение оптимизирующей последовательности допустимых решений* задачи: выбирается одно из допустимых решений $\bar{X}^{(0)}$, называемое начальным приближением, и на его базе строится последовательность допустимых решений $\bar{X}^{(0)}, \bar{X}^{(1)}, \bar{X}^{(2)}, ..., \bar{X}^{(n)}$, где $\bar{X}^{(k+1)} = \bar{X}^{(k)} + \Delta\bar{X}^{(k)}$, k=0,1,…,n, причем $\Delta\bar{X}^{(k)}$ выбирается так, чтобы при поиске $\min_{f(\bar{X})}$ выполнялось условие $f(\bar{X}^{(k)}) > f(\bar{X}^{(k+1)})$, а при поиске $\max_{f(\bar{X})}$ – условие $f(\bar{X}^{(k)}) < f(\bar{X}^{(k+1)})$. Приращение $\Delta\bar{X}^{(k)}$ чаще всего является функцией одного или нескольких предыдущих членов последовательности: $\Delta\bar{X}^{(k)} = \varphi(\bar{X}^{(k)})$ или $\Delta\bar{X}^{(k)} = \varphi(\bar{X}^{(k)}, \bar{X}^{(k-1)})$. Построение последовательности заканчивается в момент выполнения необходимых условий $\mathrm{ext}_{f(\bar{X})}$.

Процедура типа 1 применяется в вычислительной практике всегда, когда это не связано со слишком большим объемом вычислений. Применению процедуры 2 препятствует необходимость получения аналитических выражений производных целевой функции. Процедуру типа 3 использует один из методов условной оптимизации - метод штрафных функций, процедуру типа
4 - два метода одномерной оптимизации (методы Больцано и "золотого сечения"). Большинство методов оптимизации основаны на процедуре типа 5.

4.3 Методы одномерной оптимизации

К числу наиболее популярных численных методов одномерной оптимизации, т.е. поиска экстремума функции $f(x)$, относятся: метод Больцано (деления интервала пополам), метод "золотого сечения" и пошаговый метод. Первые два метода ориентированы на поиск $\mathrm{ext}\, f(x)$ внутри фиксированного интервала $(a;b)$ оси x, последний – на поиск $\mathrm{ext}\, f(x)$ в окрестности заданной точки x_0.

Будем рассматривать эти методы как методы поиска $\min f(x)$ (поиск $\max f(x)$ можно заменить поиском $\min[-f(x)]$.

Метод Больцано при поиске минимума функции $f(x)$ предусматривает следующие действия (рис. 4.6):

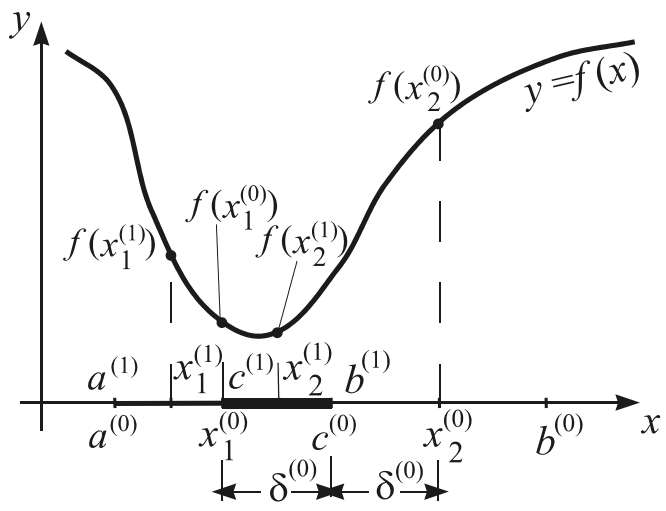


Рис. 4.6 Иллюстрация к методу Больцано

1) определяется средняя точка интервала $(a;b)$ $c = (a+b)/2$;

2) выбирается число $0 <\delta< (b-a)/2$ (наиболее популярная рекомендация: $\delta = (b-a)/4$) и определяются точки $x_1=c-\delta$ и $x_2=c+\delta$;

3) вычисляются значения функции в этих точках $f(x_1)$ и $f(x_2)$;

4) если $f(x_1) < f(x_2)$, то интервал $(a;b)$ стягивается в свою левую половину: $b \to c$, в противном случае – в правую: $a \to c$.

Для нового интервала $(a;b)$ вновь выполняются действия п.п. 1)-4). Процесс деления интервала продолжается до тех пор, пока его длина не станет меньше заданной точности: $b-a < \varepsilon$. При завершении процесса поиска за точку минимума $f(x)$ принимается середина последнего отрезка: $x^*=(a+b)/2$.

Достаточные условия сходимости алгоритма метода Больцано:

а) функция $f(x)$ непрерывна внутри интервала $(a;b)$ оси x;

б) $f(x)$ унимодальна на интервале $(a;b)$, т.е. имеет внутри него единственный экстремум;

в) в некоторой окрестности искомой точки x^* $f(x)$ является монотонной (с одной стороны возрастает, с другой - убывает).

При тех же условиях сходится алгоритм метода "золотого сечения".

Определение: "Золотым сечением" отрезка называется его деление на две части таким образом, что отношение длины отрезка к его большей части равно отношению большей части к меньшей.

Следовательно, для отрезка единичной длины: $1/t = t/(1-t) \to t^2+t-1=0$, откуда $t = -\frac{1}{2} \pm \sqrt{\left(\frac{1}{4}+1\right)}$; $|t| < 1 \to t = \frac{\sqrt{5}-1}{2} = 0.618,\ 1-t = \frac{3-\sqrt{5}}{2} = 0.382.$

Алгоритм *метода "золотого сечения"* при поиске минимума функции $f(x)$ включает операции (рис. 4.7):

1) деление интервала $(a;b)$ точками x_1, x_2 в отношении "золотого сечения": $x_1=a+(b-a)\cdot(3-\sqrt{5})/2$, $x_2=b-(b-a)\cdot(3-\sqrt{5})/2$;

2) вычисление значений функции $f(x_1)$ и $f x_2)$;

3) при $f(x_1) < f(x_2)$ – отсечение от интервала $(a;b)$ его правой части: $b \to x_2$, $x_2 \to x_1$; в противном случае – левой: $a \to x_1$, $x_1 \to x_2$;

4) если $b \to x_2$, то определяется точка x_1 нового интервала $(a;b)$, а если $a \to x_1$, то точка x_2, по правилам п.1).

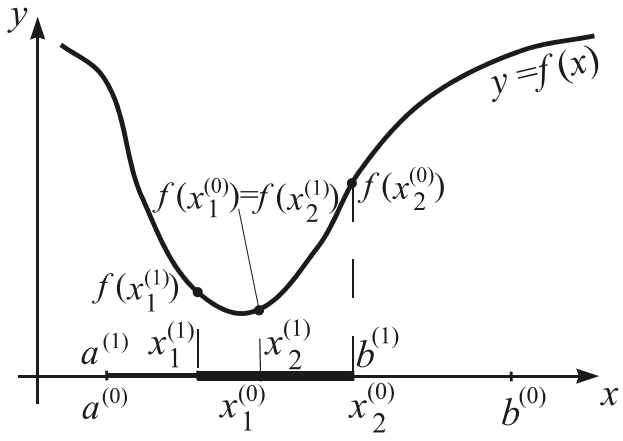


Рис. 4.7 Иллюстрация к методу "золотого сечения"

Для нового интервала $(a;b)$ вновь выполняются действия п.п. 2)-4), причем в п.2) значение функции $f(x)$ вычисляется один раз: только для вновь определяемой точки x_1 или x_2. Процесс деления интервала продолжается до тех пор, пока не выполнится условие $b - a < \varepsilon$. При завершении процесса поиска за точку минимума $f(x)$ принимается значение $x^* = (a+b)/2$.

Число модификаций исходного интервала $(a;b)$ при использовании метода “золотого сечения” больше, чем при использовании метода Больцано (от интервала отсекается не половина, а 0.382 его длины), но количество вычислений значения функции $f(x)$ существенно меньше. Поэтому в случаях, когда значение $f(x)$ вычисляется достаточно долго, метод “золотого сечения” имеет заметное преимущество перед методом Больцано.

Пошаговый метод применяется в тех случаях, когда интервал $(a;b)$ оси x, содержащий точку экстремума функции $f(x)$ неизвестен, но известно, что экстремум находится в окрестности экспериментально найденной точки x_0. Этот метод применяется на практике значительно чаще методов Больцано и "золотого сечения", т.к. условие сходимости его алгоритма намного проще: достаточно, чтобы функция $f(x)$ была непрерывна в окрестности т. x_0.

При поиске минимума $f(x)$ метод заключается в следующем (рис. 4.8):

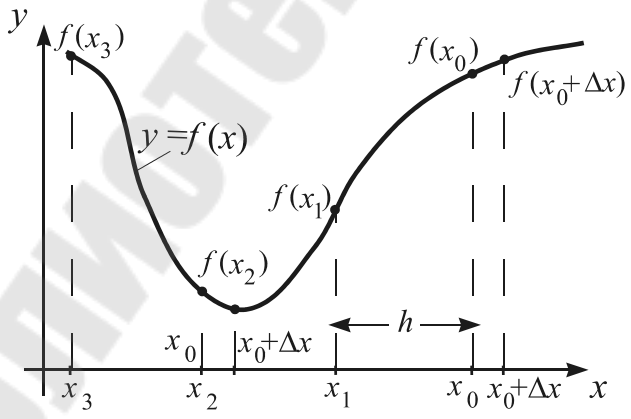


Рис.4.8 Иллюстрация к пошаговому методу

1) выполняется пробный шаг от точки x_0 с целью выбора направления поиска: $x = x_0 + \Delta x$ ($\Delta x \sim 0.5 \cdot \varepsilon$) и вычисляются значения $f(x_0)$, $f(x)$;

2) если $f(x) < f(x_0)$, то величина основного шага, с которым осуществляется движение в направлении

убывания функции, положительна ($h > 0$), в противном случае – отрицательна ($h < 0$);

3) движение в выбранном направлении с шагом h: $x_{k+1} = x_k + h$, $k = 0,1, \ldots$ осуществляется до тех пор, пока $f(x_{k+1}) < f(x_k)$;

4) если $f(x_{k+1}) \geq f(x_k)$, то при выполнении условия $h < \varepsilon$ процесс поиска заканчивается: $x^* = (x_{k+1} + x_k)/2$; если $h \geq \varepsilon$, то шаг уменьшается $h = |h|/p$, $p > 1$ (часто используют $p = e \approx 2{,}71828$) и осуществляется возврат к п. 1) с начальной точкой $x_0 = x_k$.

4.4 Методы поиска экстремума функций многих переменных

Методы поиска экстремумов функций $f(x_1,\ldots,x_n)$ подразделяются на градиентные и безградиентные по следующему признаку: градиентные основаны на вычислении и анализе частных производных функции $f(x_1,\ldots,x_n)$, безградиентные не используют значений производных.

Будем рассматривать эти методы как методы поиска $\min f(x_1,x_2,\ldots,x_n)$. Вначале рассмотрим некоторые градиентные методы.

Замечание. В практических задачах найти значения производных целевых функций вида $f(x_1,\ldots,x_n)$ аналитически, как правило, не удается и их вычисляют приближенно:

$$\frac{\partial f(x_1, x_2, \ldots, x_n)}{\partial x_i} \approx \frac{f(x_1, x_2, \ldots, x_i + \delta x_i, \ldots, x_n) - f(x_1, x_2, \ldots, x_i, \ldots x_n)}{\delta x_i}.$$

Выбор величин приращений по координатам δx_i, $i = 1,2,\ldots,n$ зависит от возможностей используемой ЭВМ и необходимой точности вычислений.

4.4.1 Метод координатного спуска

Идея метода: Движение от начальной точки по направлению одной из осей координат до момента начала возрастания целевой функции, переход к направлению другой оси и т.д., пока не будет достигнута точка, движение из которой по любой оси координат с минимально возможным шагом приводит к увеличению значения целевой функции (рис. 4.9).

Основные этапы поиска $\min f(x_1,\ldots,x_n)$ методом координатного спуска:

1) выбор начального приближения $(x_1^{(0)}, x_2^{(0)}, \ldots, x_n^{(0)})$;

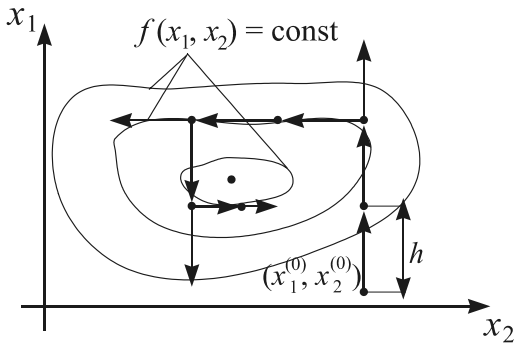


Рис. 4.9 Иллюстрация к методу координатного спуска

2) выбор направления поиска, т.е. номера $i \in (1,2,...,n)$ компоненты вектора $(x_1,x_2,...,x_n)$, которая будет измененяться;

3) вычисление значения производной целевой функции по выбранному аргументу $f_i' = \frac{\partial f(x_1^{(0)}, x_2^{(0)}, ..., x_n^{(0)})}{\partial x_i}$ (если $f_i' > 0$, то с ростом x_i значение $f(x_1, x_2,..., x_n)$ увеличивается, а если $f_i' < 0$, то уменьшается);

4) изменение значений $x_1,x_2,...,x_n$ в соответствии с выражением

$$x_i^{(k+1)} = x_i^{(k)} - h \cdot \text{sign}\left(\frac{\partial f(x_1^{(k)}, x_2^{(k)}, ..., x_n^{(k)})}{\partial x_i} \right); \; i \in (1,2,...,n); \; k = 0,1,2,...;$$
$$x_j^{(k+1)} = x_j^{(k)}; \; j = 1,2,...,i-1,i+1,...,n \qquad (4.3)$$

до тех пор, пока $f(x_1^{(k+1)},x_2^{(k+1)},...,x_n^{(k+1)}) < f(x_1^{(k)},x_2^{(k)},...,x_n^{(k)})$; в противном случае производится возврат на п. 2) и выбор следующего направления поиска, при этом $x_i^{(0)} = x_i^{(k)}$, $i=1,2,...n$ (h – шаг поиска, sign(z) – знак выражения (z);

5) если попытка движения с шагом h в любом из n возможных направлений приводит к ситуации $f(x_1^{(k+1)},x_2^{(k+1)},...,x_n^{(k+1)}) \geq f(x_1^{(k)},x_2^{(k)},...,x_n^{(k)})$, то осуществляется дробление шага: $h = h/p$ ($p > 1$) и вновь выполняется п. 4);

6) поиск считается законченным, когда значение h становится меньше заданной точности ε.

4.4.2 Методы градиента

Определение: Градиент функции $f(x_1,x_2,...,x_n)$ в точке $(x_1^{(0)},x_2^{(0)},...,x_n^{(0)})$ – это вектор, длина которого

$$\left| grad \; f\left(x_1^{(0)}, x_2^{(0)}, ..., x_n^{(0)}\right) \right| = \sqrt{\sum_{i=1}^{n} \left(\frac{\partial f\left(x_1^{(0)}, x_2^{(0)}, ..., x_n^{(0)}\right)}{\partial x_i} \right)^2} \qquad (4.4)$$

характеризует скорость возрастания функции в этой точке, а направление соответствует направлению быстрейшего возрастания функции. Антиградиент - это вектор такой же длины, направленный в противоположную сторону (рис. 4.10).

Идея методов: Каждая следующая точка поиска $\min f(x_1,x_2,...,x_n)$

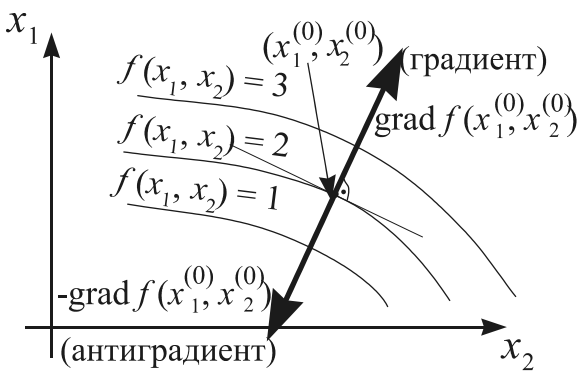


Рис.4.10 Градиент и антиградиент функции $f(x_1,x_2)$

(каждый новый член минимизирующей последовательности) получается в результате перемещения из предыдущей точки по направлению антиградиента целевой функции. Если в результате этого перемещения наблюдается увеличение значения целевой функции, то значение рабочего шага поиска h уменьшается. Поиск прекращается, когда выполняется необходимое условие ext f $(x_1,x_2,...,x_n)$, например длина градиента становится меньше требуемой точности:

$$\sqrt{\sum_{i=1}^{n}\left(\frac{\partial f\left(x_1^{(k)},x_2^{(k)},...,x_n^{(k)}\right)}{\partial x_i}\right)^2} < \varepsilon, \qquad (4.5)$$

либо меньше требуемой точности становится величина шага поиска:

$$h < \varepsilon . \qquad (4.6)$$

Различают методы градиента с переменным шагом и с постоянным шагом (рис. 4.11). При использовании метода градиента *с переменным шагом* изменение значений $x_1,x_2,...,x_n$ производится согласно выражению

$$x_i^{(k+1)} = x_i^{(k)} - h \cdot \frac{\partial f(x_1^{(k)}, x_2^{(k)},..., x_n^{(k)})}{\partial x_i}, \quad i=1,2,...,n, \quad k=0,1,2\ldots, \qquad (4.7)$$

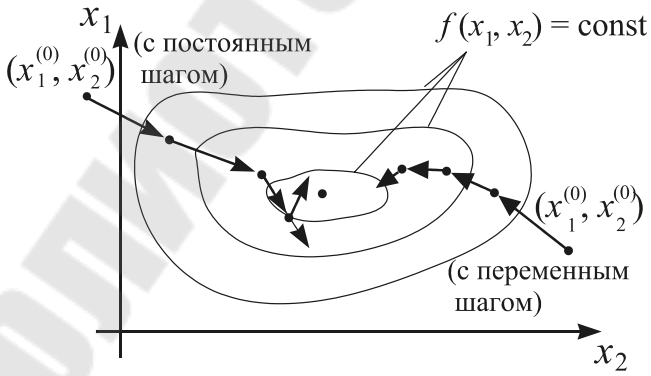


Рис. 4.11 Иллюстрация к методам градиента

а останов поиска - при выполнении неравенства (4.5). При возникновении ситуации f $(x_1^{(k+1)},x_2^{(k+1)},...,x_n^{(k+1)}) \geq f$ $(x_1^{(k)}, x_2^{(k)},..., x_n^{(k)})$ значение h уменьшается, например делится на число $p > 1$. Характер изменения

значений $x_1,x_2,...,x_n$
согласно (4.7) зависит от величины и знака соответствующих частных производных целевой функции. По мере приближения к точке $\min f(x_1,x_2,...,x_n)$ абсолютные величины частных производных уменьшаются, следовательно и шаг поиска является переменным – уменьшается по мере приближения к искомой точке. Такой характер поиска $\min f(x_1,x_2,...,x_n)$ требует иногда весьма значительных затрат времени.

Применение метода градиента *с постоянным шагом* позволяет сократить затраты времени, но требует несколько большего объема вычислений при изменении значений аргументов целевой функции. Его основное соотношение:

$$x_i^{(k+1)} = x_i^{(k)} - h \cdot \frac{\dfrac{\partial f(x_1^{(k)},x_2^{(k)},...,x_n^{(k)})}{\partial x_i}}{\sqrt{\sum_{j=1}^{n}\left(\dfrac{\partial f(x_1^{(k)},x_2^{(k)},....,x_n^{(k)})}{\partial x_j}\right)^2}},$$

$$i=1,2,...,n; \quad k=0,1,2,... , \tag{4.8}$$

т.е. расстояние между точками $(x_1^{(k)},x_2^{(k)},...,x_n^{(k)})$ и $(x_1^{(k+1)},x_2^{(k+1)},...,x_n^{(k+1)})$ равно

$$\sqrt{\sum_{j=1}^{n}\left(x_j^{(k+1)}-x_j^{(k)}\right)^2} = \sqrt{h^2 \cdot \sum_{j=1}^{n}\left(\frac{\partial f(x_1^{(k)},...,x_n^{(k)})}{\partial x_j}\right)^2 \Bigg/ \sum_{j=1}^{n}\left(\frac{\partial f(x_1^{(k)},...,x_n^{(k)})}{\partial x_j}\right)^2} = h,$$

следовательно, величина шага поиска в данном случае постоянна и равна выбранному значению h. Если изменение аргументов целевой функции в соответствии с (4.8) приводит к увеличению ее значения, шаг поиска уменьшается. Останов поиска $\min f(x_1,x_2,...,x_n)$ по методу градиента с постоянным шагом осуществляется при выполнении неравенства (4.6).

4.4.3 Метод наискорейшего спуска

Так называют модификацию метода градиента с постоянным шагом, позволяющую сократить общий объем вычислений при некотором увеличении числа членов минимизирующей последовательности за счет меньшего количества вычислений частных производных целевой функции. При использовании этого метода аргументы целевой функции изменяются в соответствии с выражением (4.8), но значения ее производных не пересчитываются до тех пор, пока

не сложится ситуация $f(x_1^{(k+1)},x_2^{(k+1)},...,x_n^{(k+1)}) \geq f(x_1^{(k)},x_2^{(k)},...,x_n^{(k)})$ (рис. 4.12). Дробление шага поиска производится, когда во вновь выбранном направлении (после пересчета значений частных производных) не удается сделать ни одного результативного шага, останов поиска – при выполнении неравенства (4.6).

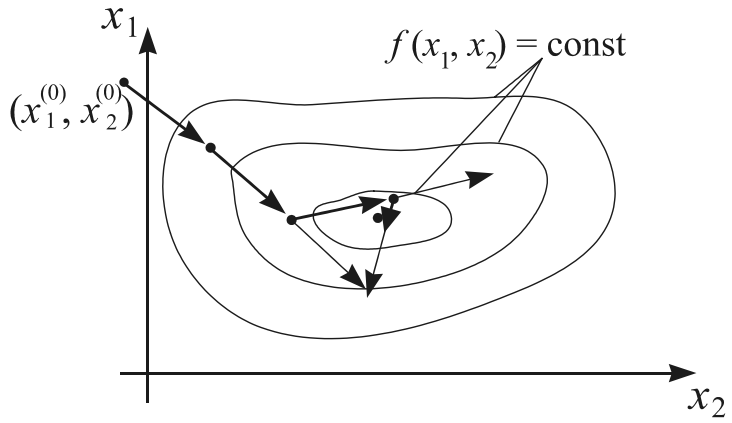


Рис. 4.12 Иллюстрация к методу наискорейшего спуска

Основные этапы поиска $\min f(x_1,x_2,...,x_n)$ методом наискорейшего спуска:

1) выбор начального приближения $(x_1^{(0)},x_2^{(0)},...,x_n^{(0)})$;

2) определение значений частных производных $f(x_1,x_2,...,x_n)$ в этой точке;

3) изменение значений x_i, $i=1,2,...,n$ в соответствии с выражением (4.8) до момента начала возрастания целевой функции без пересчета ее частных производных;

4) если ситуация $f(x_1^{(k+1)},x_2^{(k+1)},...,x_n^{(k+1)}) \geq f(x_1^{(k)},x_2^{(k)},...,x_n^{(k)})$ возникает при $k > 0$, то начальным приближением становится предыдущая точка: $x_i^{(0)} = x_i^{(k)}$, $i=1,2,...,n$ и вновь выполняются п.п. 2), 3);

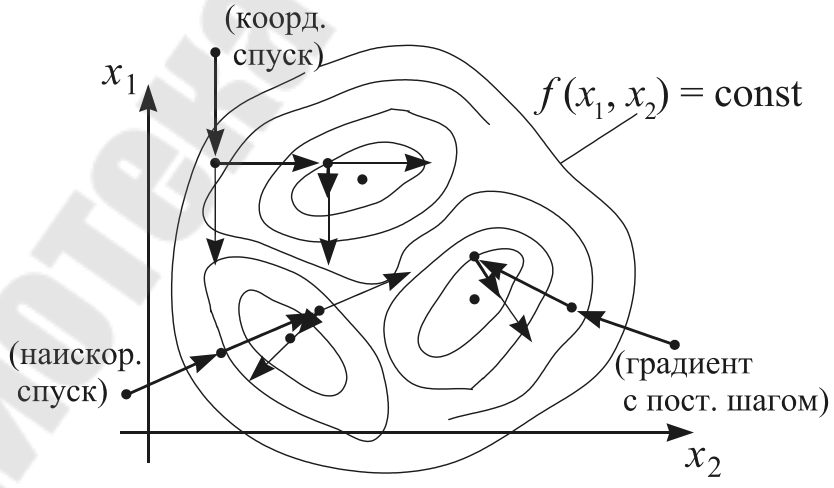


Рис. 4.13 Поиск локальных экстремумов многоэкстремальной функции

5) если $f(x_1^{(k+1)},x_2^{(k+1)},...,x_n^{(k+1)}) \geq f(x_1^{(k)},x_2^{(k)},...,x_n^{(k)})$ уже при $k = 0$, то

осуществляется дробление шага $h=h/p$ ($p > 1$); при $h \geq \varepsilon$ (заданная точность) выполняется п. 3), иначе поиск заканчивается, $x_i^* = x_i^{(k)}$, $i=1,2,...,n$.

Рассмотренные методы поиска экстремума функций многих переменных носят общее название: *градиентные методы первого порядка* (порядок метода равен наивысшему порядку производной целевой функции, участвующей в вычислениях). Им свойственны следующие общие недостатки:

1) Нахождение локального экстремума целевой функции, а не глобального (рис. 4.13). Это недостаток абсолютного большинства методов решения оптимизационных задач. Его можно устранить, если удается обосновать выбор начального приближения, находящегося вблизи глобального экстремума.

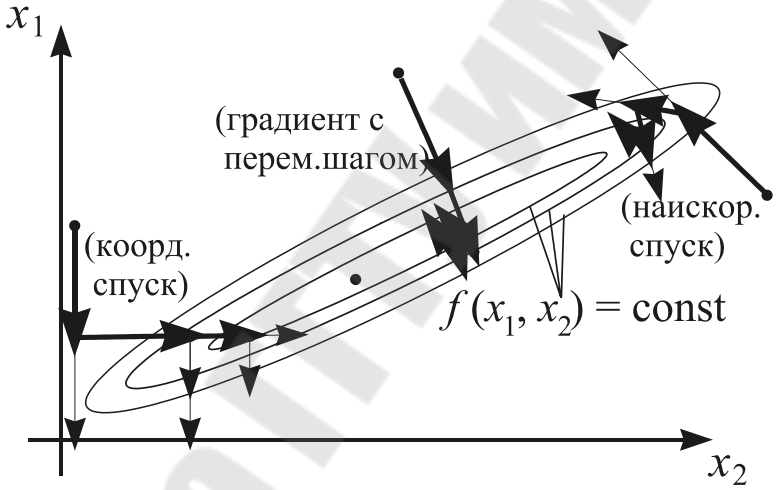


Рис. 4.14 Поиск минимума функции, имеющей "овраг"

2) Использование значений частных производных целевой функции. Это, с одной стороны, увеличивает объем вычислений (количество вычислений значений целевой функции), а с другой – увеличивает погрешность решения, т. к. производные чаще всего вычисляются через разностные отношения.

3) "Застревание в овраге" целевой функции, т.е. в области значений x_i, i =1,2,...,n, в которой значения f (x_1,x_2,...,x_n) почти не меняются (рис. 4.14).

Градиентные методы с остановкой по условию (4.6) “застревают в овраге”, т.е. наблюдается иллюзия достижения минимума. Если же в качестве условия останова используется длина градиента, то поиск в “овраге” будет продолжаться бесконечно долго: значения частных

производных целевой функции на “дне оврага” достаточно велики, но продвижения к точке минимума функции почти нет.

Замечание: Если результирующие точки поиска $\min f(x_1,x_2,...,x_n)$ с различных, достаточно далеко отстоящих друг от друга начальных точек не совпадают, а значения функции в них близки, значит она имеет "овраг", а если значения функции отличаются существенно, значит она имеет несколько экстремумов.

Перейдем к рассмотрению безградиентных методов поиска $\min f(x_1,x_2,...,x_n)$ (их также называют методами нулевого порядка). Можно выделить две группы этих методов. Первые требуют предварительного определения множества допустимых значений аргументов и используют стратегию его перебора (метод сеток). Общая схема вторых (методы случайных направлений и многогранника) предусматривает построение оптимизирующей последовательности значений аргументов целевой функции.

4.4.4 Метод сеток (сравнения значений функции на сетке значений аргументов)

Схему работы метода иллюстрирует рис. 4.15. Отрезки $[a_i;b_i]$, $i=1,2,...,n$, определяющие область поиска минимума функции $f(x_1,x_2,...,x_n)$, делятся на равные части длиной $h_i=(b_i-a_i)/n_i$. Значения n_i подбираются так, чтобы обеспечить одинаковый порядок чисел h_i, $i=1,2,...,n$. Во всех "узлах" полученной сетки, т.е. в точках $(a_1+i\cdot h_1, a_2+j\cdot h_2, ..., a_n+k\cdot h_n)$, $i=0,1,...,n_1$, $j=0,1,...,n_2$, ..., $k=0,1,...,n_n$, вычисляются значения функции и выбирается "узел" сетки $(x_1^*,x_2^*,...,x_n^*)$, которому соответствует минимальное значение. Если этот "узел" лежит на границе заданной области, то положение границ изменяется и описанная процедура повторяется до тех пор, пока он не станет внутренним. Если $\max\limits_{i=1,2,...,n}\{h_i\}>\varepsilon$

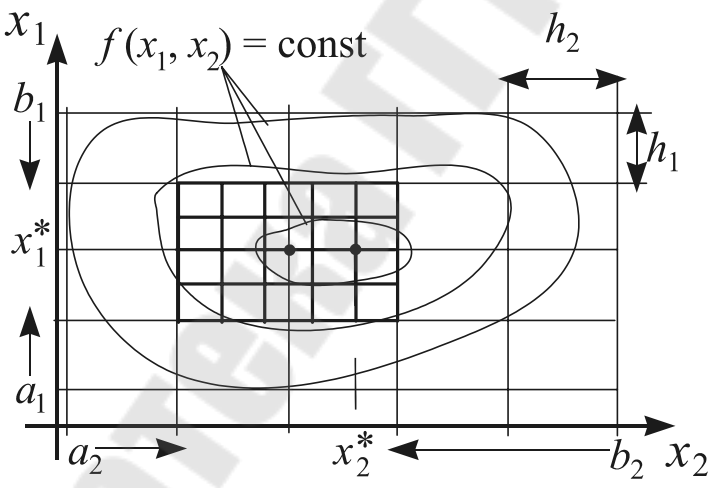


Рис. 4.15 Иллюстрация к методу сеток

(заданной точности), то вокруг этого "узла" формируется новая область: $a_i=x_i^*-h_i$, $b_i=x_i^*+h_i$, $i=1,2,...,n$, – вычисляются новые значения h_i и т.д. В противном случае за точку минимума функции принимается $(x_1^*,x_2^*,...,x_n^*)$.

Главной проблемой при использовании метода сравнения значений целевой функции на сетке значений аргументов является выбор значений n_i, при которых, с одной стороны, исключается потеря точки экстремума между "узлами" слишком крупной сетки, а с другой - обеспечивается приемлемый объем вычислений. Применение этого метода связано с большим объемом вычислений, однако при правильном выборе значений n_i он гарантирует нахождение глобального экстремума функции в заданной области.

4.4.5 Метод случайных направлений

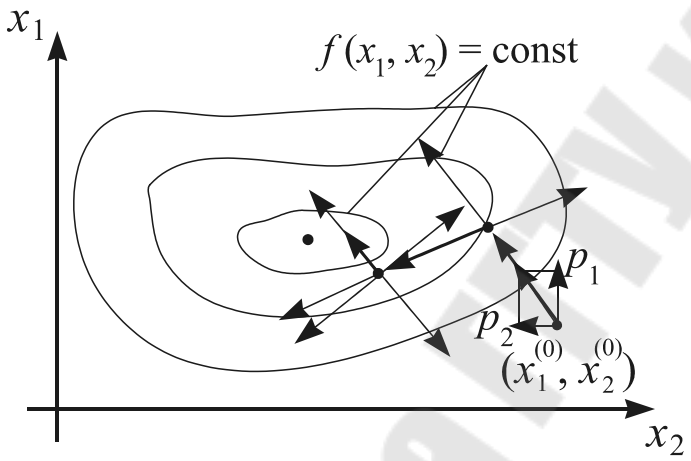


Рис. 4.16 Иллюстрация к методу случайных направлений

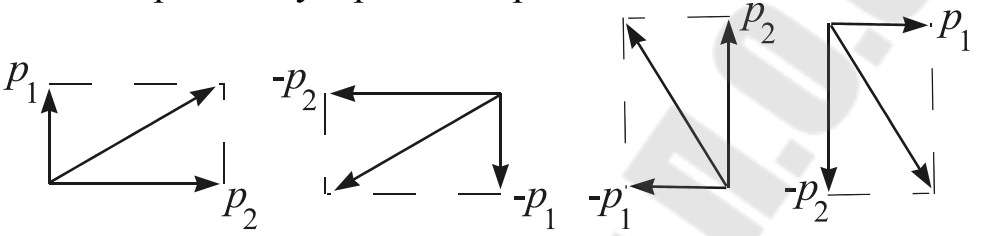
Случайный выбор направления в системе координат x_1,x_2 обеспечивается использованием в качестве приращений значений аргументов (p_1, p_2 на рис. 4.16) случайных чисел. Величина шага в выбранном таким образом направлении будет единичной, если разделить p_1 и p_2 на $P=\sqrt{p_1^2+p_2^2}$.

При поиске min f $(x_1,x_2,...,x_n)$ метод случайных направлений включает определение начальной точки поиска $\left(x_1^{(0)},x_2^{(0)},...,x_n^{(0)}\right)$ и величины рабочего шага h, выбор случайных чисел $p_1, p_2,\ldots, p_n$ и изменение значений аргументов целевой функции по правилу:

$$x_i^{(k+1)}=x_i^{(k)}-h\cdot p_i\Big/\sqrt{p_1^2+p_2^2+...+p_n^2},\quad i=1,2,\ldots n,\quad k=0,1,\ldots \qquad (4.9)$$

Если выполняется неравенство $f(x_1^{(k+1)},x_2^{(k+1)},...,\ x_n^{(k+1)})<f(x_1^{(k)},x_2^{(k)},...,\ x_n^{(k)})$ то движение в выбранном направлении с шагом h продолжается. Если неравенство не выполняется после второго, третьего и т.д. шагов, то определяется новое случайное направление и движение продолжается без изменения

величины шага. Если неудачным оказывается первый же шаг в выбранном направлении, то путем изменения знаков чисел p_i, i=1,2,...,n оно меняется на противоположное, а при повторении ситуации – на новое случайное направление. Если ситуация $f(x_1^{(k+1)},...,x_n^{(k+1)}) \geq f(x_1^{(k)},...,x_n^{(k)})$ складывается после первого же шага в любом из заданного числа случайных направлений, то величина шага поиска h уменьшается. Для функции f (x_1,x_2) обычно достаточно перебрать четыре взаимно перпендикулярных направления:



исходное противоположное перпендикулярные

Поиск закончится, когда значение h станет меньше заданной точности.

Характер движения от начальной точки к точке минимума функции при использовании метода случайных направлений не зависит от особенностей функции. Случайный выбор направления, как правило, не обеспечивает кратчайшего пути к искомой точке, но может привести к уменьшению общего объема расчетов за счет единственного вычисления целевой функции на каждом шаге поиска.

4.4.6 Метод многогранника

При поиске минимума функции двух переменных метод предусматривает формирование на плоскости x_1,x_2 правильного треугольника и определение значений функции f (x_1,x_2) в точках плоскости, соответствующих его вершинам. Вершина, которой отвечает наибольшее значение функции (вершина 1 на рис. 4.17) исключается, а новая образуется в результате симметричного отражения исключаемой через центр противоположной грани

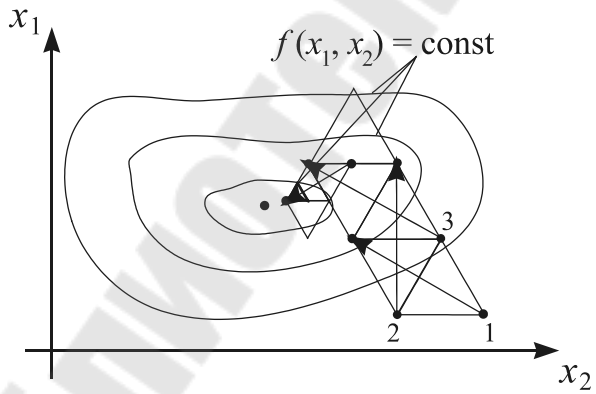


Рис. 4.17 Иллюстрация к методу многогранника

треугольника: $x_1^{(i)} = x_1^{(1)} + x_1^{(2)} + x_1^{(3)} - 2 \cdot x_1^{(i)}$,

$x_2^{(i)} = x_2^{(1)} + x_2^{(2)} + x_2^{(3)} - 2 \cdot x_2^{(i)}$, $i \in (1,2,3)$.

Процесс перемещения треугольника по плоскости x_1,x_2 путем изменения положения одной из его вершин продолжается до момента, когда значение функции во вновь образованной вершине оказывается наибольшим. В этом случае длина ребра треугольника уменьшается, причем неподвижной остается вершина (k), которой соответствует наименьшее значение функции:

$$x_1^{(j)} = \frac{x_1^{(j)} + x_1^{(k)}}{S}, \; x_2^{(j)} = \frac{x_2^{(j)} + x_2^{(k)}}{S}, \; j \neq k, \; S > 1.$$

Поиск прекращается в момент выполнения неравенства

$$\max_{i=1,2,3; i \neq k} \left\{ \left| x_1^{(k)} - x_1^{(i)} \right|, \left| x_2^{(k)} - x_2^{(i)} \right| \right\} < \varepsilon .$$

За точку $\min f(x_1,x_2)$ принимается лучшая вершина последнего треугольника.

В случае поиска $\min f(x_1,x_2,...,x_n)$ в n-мерном пространстве формируется выпуклый многогранник, имеющий (n+1) вершин и столько же граней. Если $\max_{k=1,2,...n+1} \left\{ f(x_1^{(k)}, x_2^{(k)}, ..., x_n^{(k)}) \right\} = f(x_1^{(j)}, x_2^{(j)}, ..., x_n^{(j)})$, то положение новой j-ой вершины определяется по правилу:

$$x_i^{(j)} = \frac{2}{n} \cdot \sum_{k=1}^{n+1} x_i^{(k)} - \left(1 + \frac{2}{n} \right) \cdot x_i^{(j)}, \; i=1,2,...,n. \qquad (4.10)$$

В ситуации, когда вновь образованной вершине соответствует максимальное значение целевой функции, многогранник деформируется - положение j-ой вершины может быть определено по формуле:

$$x_i^{(j)} = \frac{3}{2 \cdot n} \cdot \sum_{k=1}^{n+1} x_i^{(k)} - \left(\frac{1}{2} + \frac{3}{2 \cdot n} \right) \cdot x_i^{(j)}, \; i=1,2,...,n, \qquad (4.11)$$

где k - вершина, которой соответствует наименьшее значение функции.

Поиск прекращается при выполнении неравенства:

$$\max_{j=1,...n+1; j \neq k} \left\{ \left| x_i^{(j)} - x_i^{(k)} \right| \right\} < \varepsilon, \; i=1,2,...,n. \qquad (4.12)$$

Применение метода многогранника, как и метода случайных направлений, связано с вычислением единственного значения целевой функции на каждом шаге поиска, но характер движения обычно обеспечивает более короткий путь и меньший объем вычислений. Доказано, что при использовании правильного многогранника достаточно малых размеров направление движения совпадает с направлением антиградиента целевой функции.

В заключение остановимся на методах поиска $\min f(x_1,x_2,...,x_n)$ при

наличии "оврагов", см. рис. 4.14.

Причина образования "оврагов" функций многих переменных - неодинаковая чувствительность целевой функции к изменению различных аргументов (например, когда в качестве критерия оптимизации используются приведенные затраты на какое-либо мероприятие или прибыль от его реализации). Довольно часто в вычислительной практике приходится сталкиваться с разными пределами изменения различных аргументов $f(x_1,x_2,...,x_n)$, например $x_1 \in [0,1;0,8]$, $x_2 \in [100;1000]$. Если при вычислении частных производных целевой функции использовать одинаковые значения δx_i, то ее чувствительность к изменению разных аргументов будет неодинаковой - возникнет ситуация, характерная для наличия "оврага".

Использовать различные значения δx_i неудобно, вместо этого рекомендуют применять процедуру *нормирования аргументов целевой функции*: $z_i = (x_i - x_i^{\min})/(x_i^{\max} - x_i^{\min})$, $i=1,2,...,n$, тогда $z_i \in [0;1]$, $i=1,2,...,n$. Такая замена позволяет вычислять частные производные функции $f(x_1,x_2,...,x_n)$ с одинаковыми приращениями δz_i, $i=1,2,...,n$ и использовать в процессе поиска значения z_i вместо x_i, $i=1,2,\ldots,n$, но перед вычислением значений функции необходимо выполнять обратную процедуру: $x_i = x_i^{\min} + z_i \cdot (x_i^{\max} - x_i^{\min})$, $i=1,2,...,n$.

Если нормирование значений аргументов не помогает избежать "оврага", для поиска $\min f(x_1,x_2,...,x_n)$ используются специальные методы.

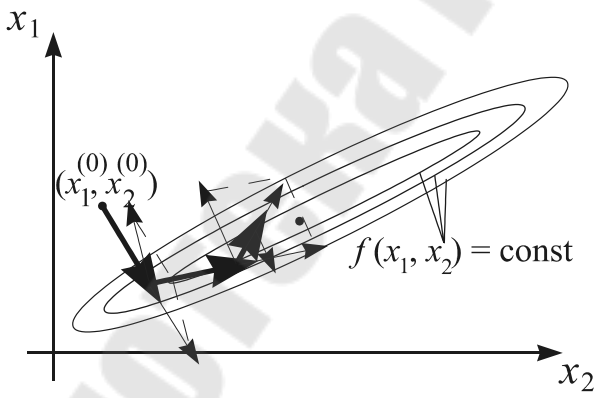


Рис. 4.18 Иллюстрация к методу сопряженных градиентов

Метод сопряженных градиентов (рис. 4.18). Это модификация метода градиента, позволяющая избежать "застревания в овраге" (в "правильном овраге", примером которого может служить эллипс с соотношением полуосей менее 1/50). Первый шаг делается в направлении антиградиента целевой функции, а второй и последующие – в направлении векторной суммы антиградиента в текущей точке и предыдущего направления. Для преодоления "застревания в овраге" через n шагов осуществляется

обновление направления: делается шаг в направлении антиградиента целевой функции. Основное соотношение метода

$$x_i^{(k+1)} = x_i^{(k)} - h \cdot p_i^{(k)}, \quad i=1,2,...,n, \tag{4.13}$$

где $$p_i^{(0)} = \frac{\partial f(x_1^{(0)}, x_2^{(0)},..., x_n^{(0)})}{\partial x_i} \Bigg/ \sqrt{\sum_{j=1}^{n}\left(\frac{\partial f(x_1^{(0)}, x_2^{(0)},..., x_n^{(0)})}{\partial x_j}\right)^2},$$

$$p_i^{(k)} = \frac{\frac{\partial f(x_1^{(k)}, x_2^{(k)},..., x_n^{(k)})}{\partial x_i}}{\sqrt{\sum_{j=1}^{n}\left(\frac{\partial f(x_1^{(k)}, x_2^{(k)}..., x_n^{(k)})}{\partial x_i}\right)^2}} + p_i^{(k-1)} \cdot \frac{\sum_{j=1}^{n}\left(\frac{\partial f(x_1^{(k)}, x_2^{(k)}..., x_n^{(k)})}{\partial x_i}\right)^2}{\sum_{j=1}^{n}\left(\frac{\partial f(x_1^{(k-1)},..., x_n^{(k-1)})}{\partial x_i}\right)^2}, \quad k=0,1,...$$

При обновлении направления второе слагаемое в последней формуле зануляется. Дробление шага и останов осуществляется аналогично методу градиента с постоянным шагом.

Метод "шагов по оврагу" (рис. 4.19). Последовательность действий при поиске минимума функции $f(x_1,x_2)$ следующая:

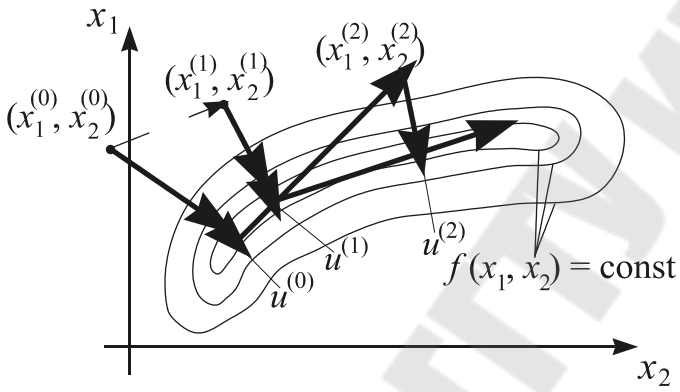


Рис. 4.19 Иллюстрация к методу "шагов по оврагу"

1) аргументы целевой функции разбиваются на две группы - существенно влияющие на ее изменение (x_1) и мало влияющие (x_2);

2) из начальной точки $\left(x_1^{(0)},x_2^{(0)}\right)$ производится поиск min $f(x_1,x_2)$ любым методом – в результате получается точка $u^{(0)}$, лежащая на дне "оврага";

3) путем изменения переменных, мало влияющих на изменение функции или случайным образом выбирается новая начальная точка $\left(x_1^{(1)},x_2^{(1)}\right)$, из которой производится поиск min $f(x_1,x_2)$ и находится точка дна "оврага" $u^{(1)}$;

4) точки $u^{(0)}$ и $u^{(1)}$ соединяются прямой и в направлении уменьшения значения функции делается "шаг по оврагу" – в результате получается новая начальная точка $\left(x_1^{(2)},x_2^{(2)}\right)$;

5) поиск min $f(x_1,x_2)$ из точки $\left(x_1^{(2)},x_2^{(2)}\right)$ даст следующую точку дна оврага $u^{(2)}$ и т.д. до тех пор, пока значение функции в точке $u^{(k+1)}$ не окажется больше, чем в точке $u^{(k)}$;

6) делается вывод, что точка минимума $f(x_1,x_2)$ находится между $u^{(k-1)}$ и $u^{(k+1)}$, поиск с точки $u^{(k-1)}$ повторяется с меньшим "шагом по оврагу", и т.д. пока этот шаг не станет меньше заданной точности.

4.5 Методы условной оптимизации

Вначале рассмотрим методы поиска $\min f(x_1,\ldots,x_n)$ при условиях (4.1).

Постановка задачи: Найти вектор $\overline{X}^* = (x_1^*, x_2^*, \ldots, x_n^*)$, доставляющий минимум функции $f(x_1,x_2,\ldots,x_n)$ при условиях $g_j(x_1,x_2,\ldots,x_n)\geq 0$, $j=1,2,\ldots,m$. Другими словами, см. рис. 4.20, требуется найти точку $(x_1^*,x_2^*,\ldots,x_n^*)$, в которой функция $f(x_1,x_2,\ldots,x_n)$ достигает минимума, но эта точка должна принадлежать области D значений $x_1,x_2,\ldots,x_n$, в которой справедливы все ограничения (4.1). Число ограничений m может быть как больше, так и меньше числа переменных n (рис. 4.21).

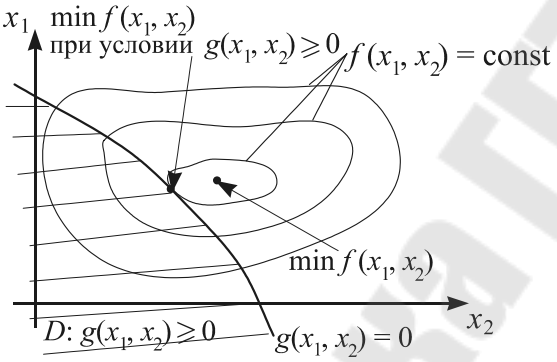


Рис. 4.20 Иллюстрация к постановке задачи условной оптимизации

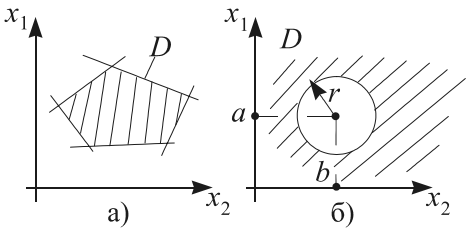


Рис. 4.21
а) пять ограничений вида $a_ix_1+b_ix_2+c_i\leq 0$
б) одно ограничение вида $(x_1-a)^2+(x_2-b)^2-r^2\geq 0$

Наиболее популярными методами поиска $\min f(x_1,x_2,\ldots,x_n)$ при условиях (4.1) являются методы штрафных функций, прямого поиска с возвратом, возможных направлений, случайных направлений, сравнения значений функции на сетке значений аргументов.

4.5.1 Метод штрафных функций

Идея метода (рис. 4.22): Наличие ограничений учитывается путем изменения целевой функции - введения в нее штрафа за нарушение ограничений. Одним из методов безусловной оптимизации осуществляется поиск минимума функции

$$F(x_1,\ldots,x_n)=f(x_1,\ldots,x_n)+\alpha\cdot Ш(x_1,\ldots,x_n), \qquad (4.14)$$

где α - положительное число, выбираемое таким образом, чтобы всюду за пределами области D выполнялось неравенство

$$\alpha\cdot\left|\frac{\partial Ш(x_1,x_2,\ldots,x_n)}{\partial x_j}\right| >> \left|\frac{\partial f(x_1,x_2,\ldots,x_n)}{\partial x_j}\right|,\ j=1,2,\ldots,n.$$

Функцию штрафа обычно записывают в виде

$$Ш(x_1,x_2,\ldots,x_n)=\sum_{j=1}^{m} q_j(x_1,x_2,\ldots,x_n), \qquad (4.15)$$

где $q_j(x_1,x_2,\ldots,x_n)=\begin{cases}\left[g_j(x_1,x_2,\ldots,x_n)\right]^2, & g_j(x_1,x_2,\ldots,x_n)<0;\\ 0, & g_j(x_1,x_2,\ldots,x_n)\ge 0.\end{cases}$.

Если ограничение одно (m=1), то необходимо найти минимум функции $F(x_1,\ldots,x_n) = f(x_1,\ldots,x_n)+ k\cdot\alpha\cdot[g(x_1,\ldots,x_n)]^2$, где

$$k=\begin{cases}1, & g(x_1,x_2,\ldots,x_n)<0\\ 0, & g(x_1,x_2,\ldots,x_n)\ge 0\end{cases},\ \alpha >> 0.$$

В области D функция $F(x_1,x_2,\ldots,x_n)$ совпадает с функцией $f(x_1,x_2,\ldots,x_n)$ и процесс поиска ее минимума протекает так же, как и при отсутствии ограничений. В момент выхода за допустимую область функция $Ш(x_1,x_2,\ldots,x_n)$ изменяет направление градиента функции $F(x_1,x_2,\ldots,x_n)$ и осуществляется возврат в допустимую область (рис. 4.23). Заметим, что возврат осуществляется не по нормали к линии ограничения, а под некоторым углом к ней в сторону уменьшения значений исходной целевой функции $f(x_1,x_2,\ldots,x_n)$.

При использовании метода штрафных функций очень важен правильный выбор значения α . При слишком малом значении α может быть найдена точка за пределами допустимой области, а при слишком большом - функция $F(x_1,x_2,\ldots,x_n)$ образует овраг вдоль поверхности $g(x_1,x_2,\ldots,x_n)$=0. Метод не чувствителен к выбору начальной точки поиска: если она окажется за пределами допустимой области, то штраф

будет включен сразу.

Наиболее популярный алгоритм метода штрафных функций предусматривает формирование функции Ш(x_1,x_2,…,x_n) согласно (2.15) в начальной точке $(x_1^{(0)},x_2^{(0)},...,x_n^{(0)})$ и использование для поиска min $F(x_1,x_2,…,x_n)$ метода градиента с постоянным шагом, где после каждого изменения значений x_1,x_2,…,x_n (см. (2.8)) вновь формируется функция Ш(x_1,x_2,…,x_n).

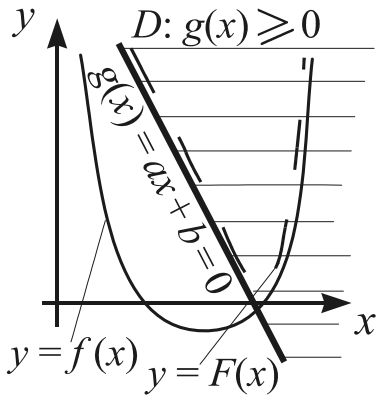


Рис. 4.22 Идея метода штрафных функций

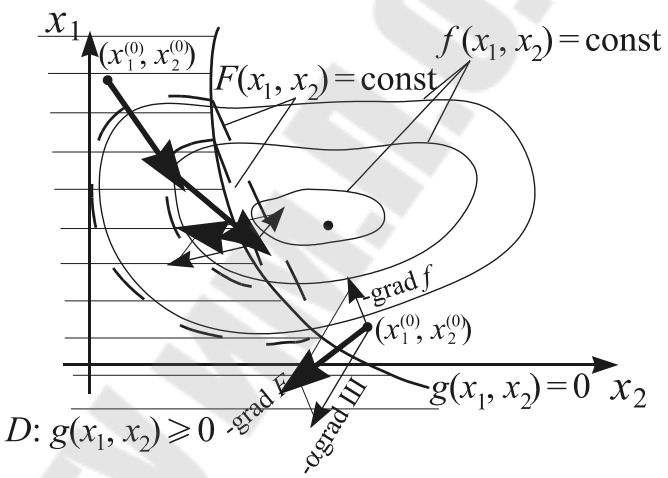


Рис. 4.23 Иллюстрация к алгоритму метода штрафных функций

4.5.2 Метод прямого поиска с возвратом

В области D допустимых значений аргументов поиск min $f(x_1,x_2,…,x_n)$ осуществляется любым методом безусловной оптимизации (чаще всего используют метод градиента с постоянным шагом и наискорейшего спуска).

При нарушении в ходе поиска хотя бы одного из неравенств (4.1) поиск прекращается и осуществляется возврат в область D по направлению векторной суммы градиентов соответствующих функций $g_j(x_1,x_2,...,x_n),\ \ j\in(1,2,...,m)$.

Возврат в область D выполняется по градиенту функции

$$G(x_1,x_2,…,x_n)=\sum_{j=1}^{m} d_j(x_1,x_2,...,x_n), \qquad (4.16)$$

где $d_j(x_1, x_2, ..., x_n) = \begin{cases} g_j(x_1, x_2, ..., x_n), & g_j(x_1, x_2, ..., x_n) < 0; \\ 0, & g_j(x_1, x_2, ..., x_n) \geq 0; \end{cases}$, т.е. значения параметров задачи изменяются следующим образом:

$$x_i^{(k+1)} = x_i^{(k)} + h \cdot \frac{\partial G(x_1^{(k)}, x_2^{(k)}, ..., x_n^{(k)})}{\partial x_i} \Bigg/ \sqrt{\sum_{j=1}^{n} \left[\frac{\partial G(x_1^{(k)}, x_2^{(k)}, ..., x_n^{(k)})}{\partial x_j} \right]^2}. \qquad (4.17)$$

Здесь $(x_1^{(k)}, x_2^{(k)}, ..., x_n^{(k)})$ - точка, в которой нарушаются ограничения, h - текущее значение шага поиска в области D. Дробление шага производится, когда значение функции в допустимой области увеличивается. Признак окончания поиска - выполнение неравенства $h < \varepsilon$.

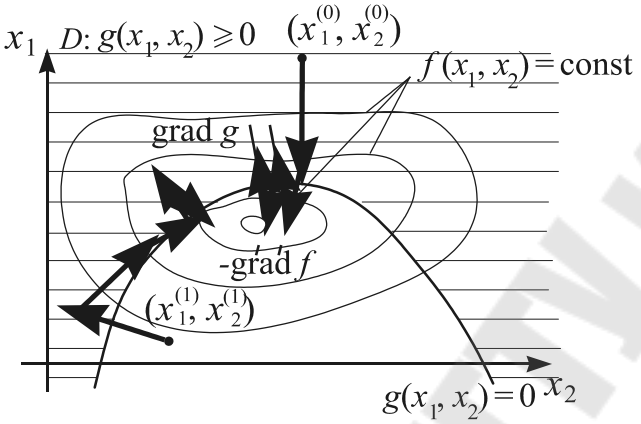


Рис. 4.24 Иллюстрация к методу прямого поиска с возвратом

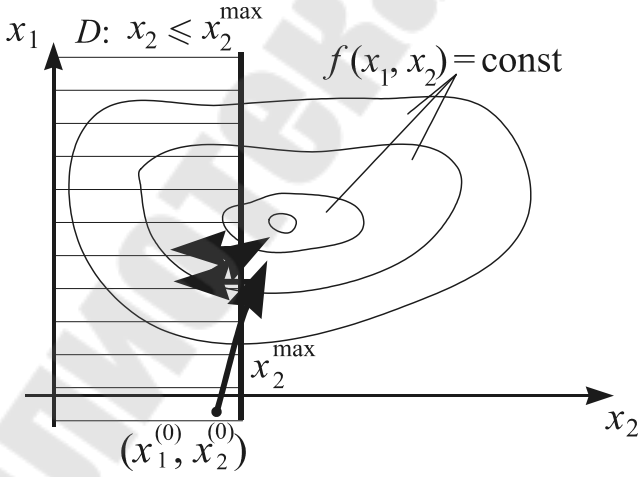


Рис. 4.25 Прямой поиск с возвратом при ограничении $x_2 \leq x_2^{\max}$

Так же, как и метод штрафных функций, метод прямого поиска с возвратом не чувствителен к выбору начальной точки поиска: движение из точки $(x_1^{(1)}, x_2^{(1)})$ на рис. 4.24 начнется сразу с применения формулы (4.17).

На практике ограничения часто задаются в виде:

$$x_i^{\min} \leq x_i \leq x_i^{\max}, \; i \in (1,2,...,n)$$

.

При нарушении некоторых из них для возврата в область D по нормали к линии

ограничения нет необходимости использовать соотношение (4.17) – достаточно уменьшить или увеличить соответствующие параметры на величину шага поиска (рис. 4.25).

Поскольку возврат в область D производится по нормали к линии ограничения, этот метод проигрывает в скорости методу штрафных функций, но не связан с образованием "оврагов" целевой функции.

Алгоритм метода прямого поиска с возвратом предусматривает проверку выполнения ограничений (4.1) в начальной точке и после каждого изменения значений $x_1,x_2,\ldots,x_n$. В случае невыполнения некоторых из них согласно (4.16) формируется функция $G(x_1,x_2,\ldots,x_n)$ и значения $x_1,x_2,\ldots,x_n$ изменяются в соответствии с соотношением (4.17) до тех пор, пока не будет обеспечено выполнение всех ограничений (4.1).

4.5.3 Метод возможных направлений

Определения: а) Ω_0 – *конус допустимых направлений* поиска $\min f(x_1,x_2,\ldots,x_n)$ при условиях $g_j(x_1,x_2,\ldots,x_n)\geq 0$, $j=1,2,\ldots,m$ – все направления в окрестности текущей точки, не приводящие к выходу за область D; б) Ω_1 – *конус подходящих направлений* – все направления, вдоль которых функция $f(x_1,x_2,\ldots,x_n)$ убывает в окрестности текущей точки;

в) $\Omega_0 \cap \Omega_1$ – *конус возможных направлений* – пересечение конусов допустимых и подходящих направлений (все направления, вдоль которых функция $f(x_1,x_2,\ldots,x_n)$ убывает при выполнении ограничений).

Для любой точки $(x_1,x_2,\ldots,x_n)$, лежащей на поверхности ограничений, в центре конуса Ω_1 находится вектор $[-\operatorname{grad} f(x_1,x_2,\ldots,x_n)]$, в центре конуса Ω_0 - вектор $\operatorname{grad} G(x_1,x_2,\ldots,x_n)$ (функция $G(x_1,x_2,\ldots,x_n)$ формируется согласно (4.16)). Центру конуса возможных направлений будет соответствовать векторная сумма $[-\operatorname{grad} f(x_1,x_2,\ldots,x_n)]$ и $\operatorname{grad} G(x_1,x_2,\ldots,x_n)$, т.е. направление

$$P_i(x_1,\ldots,x_n) = -\frac{\dfrac{\partial f(x_1,\ldots,x_n)}{\partial x_i}}{\sqrt{\sum_{j=1}^{n}\left[\dfrac{\partial f(x_1,\ldots,x_n)}{\partial x_j}\right]^2}} + \frac{\dfrac{\partial G(x_1,\ldots,x_n)}{\partial x_i}}{\sqrt{\sum_{j=1}^{n}\left[\dfrac{\partial G(x_1,\ldots,x_n)}{\partial x_j}\right]^2}},\ i=1,\ldots,n \qquad (4.18)$$

В точке условного минимума функции $f(x_1,x_2,\ldots,x_n)$, см. рис. 4.26, конусы допустимых и подходящих направлений не пересекаются, т.е. $\Omega_0 \cap \Omega_1 = \varnothing$: векторы $[-\text{grad}\, f(x_1,x,\ldots,x_n)]$ и $\text{grad}G(x_1,x_2,\ldots,x_n)$ лежат на одной прямой и направлены в противоположные стороны.

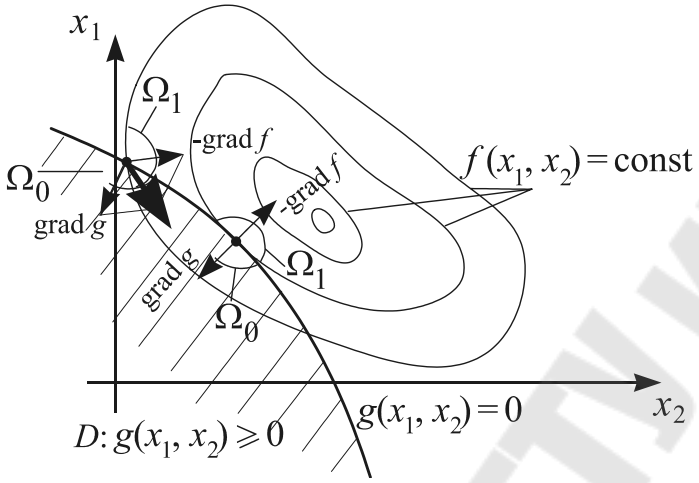


Рис. 4.26 Иллюстрация к методу возможных направлений

Алгоритм поиска условного минимума функции $f(x_1,x_2,\ldots,x_n)$ методом возможных направлений сводится к следующему: из начальной точки внутри области D осуществляется поиск $\min f(x_1,x_2,\ldots,x_n)$ любым методом безусловной оптимизации; при выходе за пределы области D поиск с предыдущей точки ведется в направлении, определяемом формулой (4.18); дробление шага поиска осуществляется, когда движение в центре конуса возможных направлений приводит к возрастанию целевой функции, либо когда первый же шаг в этом направлении приводит к нарушению ограничений; окончания поиска – при выполнении неравенства $h < \varepsilon$.

.Величина шага поиска в направлении, определяемом (4.18), должна быть постоянной (равной h), т.е. значения переменных $x_1,x_2,\ldots,x_n$ следует изменять в соответствии с выражением

$$x_i^{(k+1)} = x_i^{(k)} + h \cdot P_i(x_1^{(k)},\ldots,x_n^{(k)}) \Big/ \sqrt{\sum_{j=1}^{n}\left[P_i(x_1^{(k)},\ldots,x_n^{(k)})\right]^2},\ i=1,\ldots,n;\ k=0,1,\ldots, \qquad (4.19)$$

иначе возможны ситуации, когда величина шага поиска h значительна, а движения в конусе возможных направлений почти нет.

Метод возможных направлений чувствителен к выбору начальной точки поиска - за пределами области допустимых значений параметров $x_1,x_2,\ldots,x_n$ возможные направления отсутствуют и алгоритм метода неработоспособен.

Метод случайных направлений и метод сеток, используемые для решения задач на условный экстремум, почти полностью подобен методам безусловной оптимизации. При использовании первого к числу неперспективных дополнительно относятся все направления из текущей точки, которые приводят к нарушению хотя бы одного ограничения $g_j(x_1,x_2,...,x_n)\geq 0,\quad j\in(1,2,...,m)$, а при использовании второго значение целевой функции вычисляется только в тех точках $(x_1,x_2,\ldots,x_n)$, для которых выполняются все ограничения.

Методика поиска экстремума функции многих переменных при наличии связей состоит в том, чтобы найти вектор $X^* = (x_1^*,x_2^*,\ldots,x_n^*)$, доставляющий минимум функции $f(x_1,x_2,\ldots,x_n)$ при условиях (4.2), т.е.

$$h_k(x_1,x_2,...,x_n)=0,\ k=1,2,\ldots,p.$$

При решении подобных задач связи $h_k(x_1,x_2,...,x_n)=0$, k=1,…,p заменяются ограничениями (см. рис. 4.27)

$$\begin{matrix} h_k(x_1,...,x_n)+\delta_k\geq 0 \\ h_k(x_1,...,x_n)-\delta_k\leq 0 \end{matrix}, \qquad k=1,\ldots,p, \qquad (4.20)$$

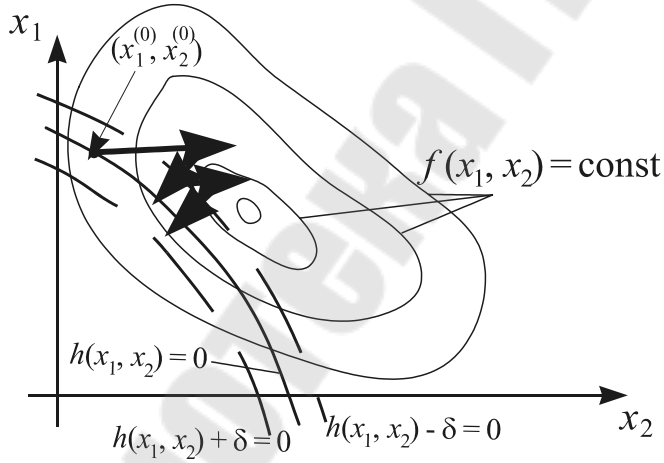


Рис. 4.27 Поиск min f (x_1,x_2) при условии $h(x_1,x_2)$=0

и задача решается методом штрафных функций, прямого поиска с возвратом или возможных направлений. Чем меньше значения δ_k, тем ближе решение новой задачи к исходной, но сложнее поиск. Обычно задают начальные значения δ_k, k=1,2,…, p (не слишком маленькие), находят решение новой задачи, а затем постепенно уменьшают δ_k, пока они не станут меньше заданной точности. Так же поступают со связями в случае решения задачи условной оптимизации

при наличии и ограничений и связей.

Для нахождения начальной точки поиска, удовлетворяющей неравенствам (4.20), рекомендуется найти значения $x_1, x_2, \ldots, x_n$, доставляющие минимум функции $\sum_{k=1}^{p}[h_k(x_1, x_2, \ldots, x_n)]^2$.

5. ЭКСПЕРИМЕНТАЛЬНОЕ МОДЕЛИРОВАНИЕ

Для построения моделей, математическое описание которых отсутствует, удобнее всего использовать методы математической статистики, так как параметры литейных процессов носят вероятностный характер. Это обусловливается двумя основными факторами. Во-первых, это недостаточно равномерное качество исходных шихтовых и формовочных материалов и значительный разброс их параметров. Во-вторых, это недостаточная точность оборудования и контрольно-измерительной аппаратуры, применяемых в настоящее время в литейных цехах, что нередко приводит к большим отклонениям технологических параметров от заданных величин.

Математическая модель процесса дает возможность количественно оценить влияние различных параметров на качественные показатели отливок и технологического процесса в целом.

Математическая модель служит базой для оптимизации технологических параметров и разработки алгоритмов управления литейными процессами, а также обеспечивает широкое применение в литейном производстве вычислительной и управляющей техники.

В последние годы в литейном производстве начали применять статистические методы планирования многофакторных экспериментов, позволяющих при сравнительно небольшом количестве опытов получить достаточно достоверную модель процесса.

Схема построения математической модели производственного процесса приведена на рис 5.1.

Математическое моделирование литейных процессов с помощью методов планирования экспериментов включает в себя следующие основные этапы.

1. Установление наиболее важных технологических факторов, влияющих на показатели процесса (например, на прочность и твердость сплава влияют химический состав металла, температура его перегрева, температура заливки, плотность и влажность формы и др.), т е выявление общих функциональных связей типа:

$$y = f\left(X_1, X_2, ..., X_m\right) \qquad (5.1)$$

где y – показатель процесса, X_i – i-й параметр процесса – изучаемый фактор.

Рис. 5.1 Схема построения математической модели литейных процессов с помощью методов планирования экспериментов

2. Выбор основного уровня и интервалов варьирования по всем изучаемым факторам–независимым переменным, которые нормируются, выражаются в относительных единицах и в процессе эксперимента при планировании первого порядка принимают значения (+1) и (–1), т е верхний и нижний уровни соответственно.

3. Планирование многофакторного (активного) эксперимента для i-го числа факторов, где i = 1,2, ..., *m.*

Планирование состоит в построении матрицы, определяющей изменения изучаемых факторов при каждом опыте. В широком смысле слова планирование эксперимента - научная дисциплина, занимающаяся разработкой и изучением оптимальных программ проведения экспериментальных исследований.

В теории планирования эксперимента часто определяют эксперимент как совокупность условий и результатов проведения серий опытов.

Формально план часто можно представить в виде последовательности векторов $\bar{x}_u$, u=1, 2, . . . , n, где n - число опытов в плане, а компоненты $\bar{x}_u$,, определяют условия каждого опыта.

В большинстве моделей, используемых в планировании эксперимента, предполагается, что факторы могут рассматриваться как детерминированные переменные. Обычно факторы выражаются в безразмерных единицах масштаба и обозначаются буквами x_i, i = 1, 2, . . ., *k.* Совокупность факторов изображается вектором $\bar{x}^{T1} = \|x_1, x_2, \ldots, x_k\|$. Здесь и далее векторы обозначаются малыми полужирными буквами, матрицы - большими полужирными.

Факторы могут различаться по числу уровней, на которых возможно их фиксировать в данной задаче. Фактор, варьируемый на p уровнях, называют p-уровневым фактором.

Основной уровень фактора, обозначаемый x_{oi}^{H} , где индекс i относится к номеру фактора, служит для фиксирования в области планирования таких условий эксперимента, которые представляют наибольший интерес для исследователя в данный момент, и относится к определенному плану эксперимента.

За единицу масштаба безразмерной системы координат принимается некоторый интервал в натуральных единицах. При нормализации фактора наряду с изменениями масштаба изменяется начало отсчета. Значение i-го фактора в безразмерной системе связано со значением этого фактора x_i^{H} в натуральной системе (в именованных единицах)

формулой

$$x_i = \frac{x_{\text{i}}^{\text{H}} - x_{\text{oi}}^{\text{H}}}{\Delta \text{x}_{\text{i}}^{\text{H}}}, \qquad (5.2)$$

где x_{oi}^{H} - основной уровень фактора, принимаемый за начало отсчета; $\Delta x_{\text{i}}^{\text{H}}$ - интервал в натуральных единицах масштаба, соответствующий одной единице масштаба в безразмерных переменных.

С геометрической точки зрения нормализация факторов равноценна линейному преобразованию пространства факторов, при котором производится перенос начала координат в точку, отвечающую основным уровням, и сжатие-растяжение пространства в направлении координатных осей.

В зависимости от наличия исходной информации о параметрах и показателях процесса применяется планирование первого или второго порядка и выбирается конкретный вид планирования. Если заранее известно, что между показателями процесса и изучаемыми факторами наблюдаются монотонные, слегка нелинейные зависимости, используют планирование первого порядка, которое предполагает постановку полного факторного эксперимента (перебирают все возможные сочетания верхнего и нижнего уровней изучаемых факторов) или дробных реплик (определенной части полного факторного эксперимента, когда ряд парных и других взаимодействий независимых переменных исключается из рассмотрения).

Если зависимости «показатели–изучаемые факторы» имеют явно выраженные экстремумы, применяют планирование второго порядка, с помощью которого в дополнение к линейным эффектам и их взаимодействиям выявляются квадратичные эффекты.

Несмотря на универсальность методов планирования экспериментов, специфика любого процесса накладывает определенные ограничения на применение этих методов. Так, например, для разных литейных процессов могут быть использованы вполне определенные типы планов экспериментов, зависящие от числа независимых переменных и необходимой точности показателей процесса. При большом числе независимых переменных и невысокой точности задания показателей планирование второго порядка, как показывает опыт, становится нерациональным, так как требует значительных затрат времени и средств В этой связи большинство литейных процессов моделируют с достаточной для практики точностью с помощью планирования первого порядка, позволяющего получить уравнения, состоящие из линейных членов и их взаимодействий, а

планирование второго порядка применяют только при небольшом числе переменных.

Успех в решении вопроса построения достоверной математической модели процесса зависит во многом от правильности выбора основного уровня и интервала варьирования для каждой независимой переменной. Интервал варьирования необходимо выбирать как можно меньшим, чтобы избежав больших погрешностей в случае наличия значительных нелинейностей в реальных характеристиках. Однако при этом значительно возрастает объем вычислений.

Поэтому рабочий диапазон изменения переменных разбивают на ряд участков, которые достаточно точно описываются линейными уравнениями. Система таких уравнений дает полную модель, справедливую для всего рабочего диапазона изменения переменных. В некоторых случаях приходится решать компромиссную задачу, в условия которой необходимо вводить требования технологического характера. Так, например, при исследовании свойств обычного серого чугуна основной уровень и интервал варьирования по химическому составу чугуна выбирают, исходя из требования, предусматривающего невозможность получения белого чугуна при низком углеродистом эквивалент и при отсутствии эффекта модифицирования.

Размах варьирования фактора указывает границы области варьирования данного фактора в данном эксперименте.

Интервал или шаг варьирования фактора, обозначаемый $\Delta x_{\text{i}}^{\text{H}}$, для фактора с номером *i* служит для перехода от натурального масштаба к безразмерному. Вместе с основным уровнем он задает область действия для данного плана, т. е. область действия есть $x_{\text{oi}}^{\text{H}} \pm \Delta x_{\text{i}}^{\text{H}}$ или иначе $(x_{\text{oi}}^{\text{H}} + \Delta x_{\text{i}}^{\text{H}}; x_{\text{oi}}^{\text{H}} - \Delta x_{\text{i}}^{\text{H}})$.

В полиномиальном уравнении регрессии эффект взаимодействия выражается параметром при членах, включающих произведения факторов. Различаются парные взаимодействия вида $x_i\, x_j$, тройные вида $x_i\, x_j\, x_k$ и более высокого порядка.

Размерность факторного пространства равна числу факторов *k*. Каждой точке факторного пространства соответствует вектор

$$\bar{x}^{T} = \left\| x_1, x_2, \ldots, x_k \right\|. \qquad (5.3)$$

Если область планирования задается интервалами возможного

изменения факторов, она представляет собой гиперпараллелепипед (в частном случае куб). Иногда область планирования задается гиперсферой.

Функция отклика выражается соотношением

$$E\{y/\bar{x}\} = \eta = f(x_1, x_2, \ldots, x_k, \Theta_1, \Theta_2, \ldots, \Theta_m) \quad (5.4)$$

или

$$E\{y/\bar{x}\} = \eta = f(\bar{x}, \bar{\Theta}). \quad (5.5)$$

Функция отклика связывает между собой математическое ожидание отклика $E\{y/\bar{x}\} = \eta$, совокупность факторов, выражаемую вектором $\bar{x}$, и совокупность параметров модели, определяемую вектором $\bar{\Theta}^{\mathrm{T}} = \|\Theta_1, \Theta_2, \ldots \Theta_m\|$.

Параметры модели априори неизвестны и подлежат определению из эксперимента.

На функцию отклика могут переноситься определения, связанные с моделью, например, линейная (по параметрам), полиномиальная, квадратичная и т. д.

Поверхность отклика имеет размерность *k* и размещена в *(k*+1)-мерном пространстве.

Параллельные опыты служат для получения выборочной оценки дисперсии воспроизводимости результатов эксперимента.

Временной дрейф обычно связывают с изменением во времени каких-либо характеристик функции отклика (параметров, положения экстремальной точки и т. п.). Различают детерминированный и случайный дрейфы. В первом случае процесс изменения параметров (или иных характеристик функции отклика) описывается детерминированной (обычно степенной) функцией времени. Во втором случае изменение параметров - случайный процесс. Если дрейф аддитивный, то поверхность отклика смещается во времени, не деформируясь (при этом дрейфует только свободный член функции отклика, т. е. член, не зависящий от значений факторов). При неаддитивном дрейфе поверхность отклика во времени деформируется. Цель планирования в условиях аддитивного дрейфа исключить влияние дрейфа на оценки эффектов факторов. При дискретном дрейфе это удается сделать путем разбиения эксперимента на блоки. При непрерывном дрейфе используют планы эксперимента, ортогональные к дрейфу, описываемому степенной функцией

известного вида.

В задачах экспериментальной оптимизации в условиях дрейфа функции отклика применяют методы адаптационной оптимизации, к которым относятся метод эволюционного планирования и последовательный симплексный метод.

Модель регрессионного анализа выражается соотношением

$$\text{у} = \text{з} + \text{е} = f(\overline{\text{х}},\overline{\text{И}}) + \text{е}, \qquad (5.6)$$

где е - случайная ошибка. Для некоторого *и*-го наблюдения имеем

$$\text{у}_{\text{и}} = \text{з}_{\text{и}} + \text{е}_{\text{у}} = f(\overline{x}_{\text{и}}, \overline{\text{И}}) + \text{е}_{\text{и}},$$

Наиболее простые предположения о случайных величинах $\varepsilon_{\text{и}}$ состоят в том, что их математические ожидания равны нулю

$$E\{\varepsilon_{\text{и}}\}=0,$$

дисперсии постоянны

$$E\{\varepsilon_{\text{и}}^2\} = \sigma^2,$$

а ковариации равны нулю

$$E\{\varepsilon_{\text{и}}\,\varepsilon_{\text{v}}\}=0,\ u \neq v.$$

Последние условия соответствуют равноточности и некоррелированности наблюдений.

Линейная по параметрам модель регрессионного анализа представима в форме

$$y = \eta + \varepsilon = \beta_1 f_1(x_1,x_2,x_3,\ldots,x_k) + \beta_2 f_2(x_1,x_2,\ldots,x_k) + \ldots + \beta_m f_m(x_1,x_2,\ldots,x_k) + \varepsilon, \qquad (5.7)$$

где β_1 - параметры модели, $i = 1, 2, \ldots, m$;

$f_1(x_1,x_2,x_k)$ - известные базисные функции переменных $x_1,x_2,\ldots,x_k$ (факторов), не зависящие от параметров модели.

Линейная модель может быть записана более лаконично

$$y = \sum_{i=1}^{m} \beta_i f_i(\overline{x}) + \varepsilon \quad \text{или} \quad y = \overline{f}^{\text{T}}(\overline{x})\overline{\beta} + \varepsilon, \qquad (5.8)$$

где $\overline{f}^{\text{T}}(\overline{x})$ -вектор-строка базисных функций (базисная вектор-функция)

$$\overline{f}^{\text{T}}(\overline{x}) = \left\| f_1(\overline{x}), f_2(\overline{x}), \ldots, f_m(\overline{x}) \right\|, \qquad (5.9)$$

β - вектор параметров модели

$$\overline{\beta} = \begin{Vmatrix} \beta_1 \\ \beta_2 \\ \cdot \\ \cdot \\ \cdot \\ \beta_m \end{Vmatrix} \qquad (5.10)$$

Модель первого порядка может содержать свободный член - дополнительный параметр; при этом обозначать параметры модели индексами, начиная с нуля $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_k x_k + \varepsilon.$

Иногда при обозначении модели первого порядка используется фиктивная переменная, тождественно равная единице: x_0=1.

С учетом этого обозначения модель может быть записана в виде суммы

$$y = \sum_{i=0}^{k} \beta_1 x_1 + \varepsilon. \qquad (5.11)$$

Модель регрессионного анализа второго порядка для факторов в общем случае содержит $\frac{(k+1)(k+2)}{2}$ параметров. Параметры модели чаще всего нумеруют не подряд от 1 до $m = \frac{(k+1)(k+2)}{2}$, а начиная с нуля и в соответствии с индексами независимых переменных, на которые умножаются параметры. Наиболее распространенная форма записи квадратичной модели следующая

$$y = \beta_0 + \beta_1 x_1 + \ldots + \beta_k x_k + \beta_{12} x_1 \cdot x_2 + \ldots + \beta_{k-1,k} \cdot x_{k-1} x_k + \beta_{11} x_1^2 + \ldots \beta_{k,k} \cdot x_k^2 + \varepsilon. \qquad (5.12)$$

Неизвестные параметры дисперсионной модели могут быть детерминированными или случайными величинами. В первом случае, модель называют моделью с постоянными факторами или моделью 1. Модель, в которой все параметры β_i (может быть за исключением одного) являются случайными величинами, называется моделью со случайными факторами или моделью II.

В промежуточных случаях модель называется смешанной.

Для проверки адекватности модели часто используют *F*-критерий Фишера.

Под коэффициентом регрессии обычно понимают параметры регрессионной модели, линейной по параметрам. Их чаще всего

обозначают буквой β.

Чтобы исключить воздействие на оценки эффектов факторов каких-либо источников неоднородности, план разбивают на блоки. Различают полноблочные планы, в которых в каждом блоке реализуется одна и та же совокупность опытов, и неполноблочные, когда блоки состоят из различных комбинаций опытов. Неполноблочные планы бывают сбалансированными и частично-сбалансированными (сбалансированные неполные блок-схемы и частично-сбалансированные неполные блок-схемы соответственно).

Точке плана с номером *u* в факторном пространстве отвечает вектор

$$\overline{x}_{\text{и}}^{\text{Т}} = \left\| x_{1\text{и}}, x_{2\text{и}}, \ldots, x_{k\text{и}} \right\|. \tag{5.13}$$

Набор основных уровней всех факторов образует вектор-точку в факторном пространстве, которая и называется центральной точкой плана:

$$\overline{x}_0^{\text{НТ}} = \left\| x_{10}^{\text{Н}}, x_{20}^{\text{Н}}, \ldots, x_{k0}^{\text{Н}} \right\|. \tag{5.14}$$

Матрица плана имеет размеры ($N \times k$), она может иметь совпадающие строки;

(i, j) - элемент матрицы плана равен уровню *j*-го фактора в *i*-м опыте.

Все строки матрицы спектра плана различны, ее размеры *(n×k),*

где *n* - число точек в спектре плана.

Матрица дублирования имеет вид

$$R = \begin{Vmatrix} r_1 & & & & 0 \\ & r_2 & & & \\ & & \cdot & & \\ & & & \cdot & \\ & & & & \cdot \\ 0 & & & & r_n \end{Vmatrix} \tag{5.15}$$

Матрица базисных функций модели состоит из *N* строк *m* столбцов. Элементами *i*-й строки такой матрицы являются значения базисных функции в *i*-м опыте.

Матрица моментов имеет размеры ($m \times m$) и может быть выражена

$$M = X^T \cdot X = \overline{X}_{\text{ус}}^{\text{Т}} R \overline{X}_{\text{ус}} \tag{5.16}$$

или

$$M = \sum_{i=1}^{N} \overline{f}(\overline{x}_i) \cdot \overline{f}^{\mathrm{T}}(x_i) = \sum_{u=1}^{n} r_u \overline{f}(\overline{x}_u) \overline{f}^{\mathrm{T}}(\overline{x}_n). \qquad (5.17)$$

В общем случае при неравноточных и коррелированных откликах матрица моментов может быть выражена:

$$M = X^{\mathrm{T}} D_y^{-1} X,$$

где D_y - ковариационная матрица вектора наблюдений.

Факторный план характеризуется наличием ряда факторов, каждый из которых варьируется на двух или более уровнях. Многие типы планов можно интерпретировать как частные случаи факторных планов.

Различают регулярные и нерегулярные дробные факторные планы (дробные реплики). Регулярность реплики означает сохранение в ее структуре некоторых важных характеристик полного плана, например, симметрии и ортогональности.

Если обозначить число символов через S, то «латинский квадрат» - это такая структура, где S символов расположены в S^2 ячейках. Символы располагаются в S строках и S столбцах так, что каждый символ встречается один и только один раз в каждой строке и в каждом столбце.

Если обозначить число символов через S, то латинский куб это такая структура, где S символов расположены в S^3 ячейках. Они располагаются в S квадратах из S строк и S столбцов так, что каждый символ встречается одинаковое число раз в квадрате.

К числу важнейших критериев оптимальности экспериментальных планов относят:

а) критерий *D*-оптимальности - это мера эффективности плана, сформулированная на языке свойств информационной матрицы плана.

Пусть $\boldsymbol{M}=\boldsymbol{X}^{\mathrm{T}}\cdot\boldsymbol{X}$ - матрица моментов плана, а $\boldsymbol{M}_{\mathrm{N}} = \frac{1}{N}\boldsymbol{X}^{\mathrm{T}}\cdot\boldsymbol{X}$ - информационная матрица плана.

Здесь N - общее число опытов в плане, $\boldsymbol{X}$ - матрица базисных функций для заданной модели и фиксированного плана, $\boldsymbol{X}^{\mathrm{T}}$ - транспонированная матрица X. Удовлетворение требования *D*-оптимальности означает минимизацию определителя матрицы $\boldsymbol{M}_{\mathrm{N}}^{-1}$ ($\boldsymbol{M}_{\mathrm{N}}^{-1}$ матрица, обратная информационной матрице $\boldsymbol{M}_{\mathrm{N}}$) на множестве

элементов x_{ij} матрицы плана, т. е. min det $\boldsymbol{M}_N^{-1}$

$x_{ij}\xi\Omega_x$.

Здесь x_{ij} - элемент *i*-й строки и *j*-го столбца матрицы плана, *i*=l, 2, . . . , *N*, *j*=1, . . . , *k (k* - число факторов). Ω_x - область экспериментирования. det - обозначение операции вычисления определителя матрицы.

D - оптимальный план минимизирует на множестве допустимых планов обобщенную дисперсию оценок коэффициентов регрессии;

б) критерий *A*-оптимальности - это мера эффективности плана, сформулированная на языке свойств информационной матрицы плана.

Пусть $\boldsymbol{M}=\boldsymbol{X}^T\cdot\boldsymbol{X}$ - матрица моментов плана, а

$\boldsymbol{M}_N = \frac{1}{N}\boldsymbol{X}^T\cdot\boldsymbol{X}$ - информационная матрица плана.

Здесь *N* - общее число опытов в плане, $\boldsymbol{X}$ - матрица базисных функций для заданной модели и фиксированного плана, $\boldsymbol{X}^T$ - транспонированная матрица $\boldsymbol{X}$. Удовлетворение требования *A*-оптимальности означает минимизацию следа матрицы $\boldsymbol{M}_N^{-1}$ на множестве элементов x_{ij} матрицы плана, т. е.

$$\min S_p \boldsymbol{M}_N^{-1},$$
$$x_{ij}\xi\Omega_x.$$

где S_p - обозначение операции вычисления следа матрицы;

x_{ij} - элемент *i*-й строки и *j*-го столбца матрицы плана, (*i*=l, 2, . . . , *N*, *j*=1, 2, . . . , *k*);

Ω_x - область экспериментирования.

A-оптимальный план минимизирует на множестве допустимых планов среднюю дисперсию оценок коэффициентов регрессии.

В настоящее время используется свыше 20 различных критериев оптимальности планов.

Планирование является ротатабельным, если матрица моментов плана инвариантна к ортогональному вращению координат.

К количественным относятся такие факторы, как температура, давление, вес и т. п. примеры качественных факторов - тип прибора, вид материала, сорт зерна и т. п. Если количественный фактор принимает в эксперименте небольшое число различных значений, то его можно рассматривать как качественный. В такой ситуации применима техника дисперсионного анализа.

6. ИСПОЛЬЗОВАНИЕ МАТЕМАТИЧЕСКОГО МОДЕЛИРОВАНИЯ В ПРОИЗВОДСТВЕННЫХ ПРОЦЕССАХ ЛИТЕЙНОГО ПРОИЗВОДСТВА

В условиях рыночной экономики для выживания предприятий большое значение имеет снижение себестоимости продукции и уменьшение сроков разработки изделия. Одним из наиболее эффективных путей достижения этих целей является автоматизация деятельности предприятия на основе современных CAD/CAM/CAE/TDM/PDM-систем, реализующих концепцию комплексной автоматизации производства, т.е. охват всего технологического цикла.

Развитие систем компьютерного моделирования литейных процессов продолжается уже более 30 лет.

Компьютерное моделирование становится неотъемлемой частью процессов конструирования новых деталей и проектирования технологических процессов их изготовления. Оно приобретает статус важного, а зачастую решающего конкурентного преимущества. Все чаще заказчики на литейную продукцию в списке требований к производителю этой продукции выдвигают требование об обязательном использовании компьютерного моделирования. Преимущества, предоставляемые системами автоматизированного моделирования литейных процессов (САМ ЛП) очевидны. В первую очередь, это возможность отработки нюансов литейной технологии на виртуальном прототипе изготавливаемой отливки, что уменьшает или полностью исключает необходимость в изготовлении пробных отливок, сокращает процесс проектирования технологии и снижает себестоимость отливки. Визуализация физических процессов литейной технологии, таких как заполнение расплавом полости литейной формы, охлаждение и затвердевание металла, его коробление под действием термических напряжений позволяет лучше понять особенности этих процессов, а следовательно, более эффективно управлять ими с целью снижения брака отливок и повышения выхода годного. Однако широкое внедрение САМ ЛП сдерживается рядом причин, среди которых недостаток информации о САМ ЛП, недостаток квалифицированных специалистов и др.

Для большинства выпускаемых отливок, ввиду их сложной геометрии, качественное решение задач расчета кристаллизации

металла возможно только с использованием численных методов, например метода конечных элементов (МКЭ) или метода конечных разностей (МКР). Численное моделирование позволяет также рассчитать процесс заполнения металлом формы, спрогнозировать возникновение дефектов.

Макро- и микропористость, эрозия формообразующей поверхности стержней и формы, засоры, ликвация и др. дефекты с достаточной точностью прогнозируются по результатам численного расчета и анализа тепловых процессов при заполнении и кристаллизации отливок.

С увеличением вычислительной мощности персональных компьютеров возникла возможность использования разработанного программного обеспечения в условиях литейного производства для отработки и оптимизации технологии литья конкретных отливок.

Однако, результаты внедрения таких пакетов программного обеспечения, как WinCast, ProCast, Flow-3d и др. литейщиками-технологами в СНГ выявили следующие проблемы:

- сложный интерфейс требует длительного периода адаптации и обучения персонала;

- трудоемкость работы по созданию расчетной модели, в частности при генерации объемной конечно-элементной (КЭ) сетки;

- длительность моделирования для конкретных отливок может составлять десятки часов, а в случае просчета нескольких вариантов технологии более одной рабочей недели;

- прогноз объемных дефектов часто не подтверждается на практике.

Важный вопрос использования CAM ЛП на производствах — это вопрос соответствия получаемых в моделировании результатов реальным производственным данным. Часто приходится сталкиваться с двумя прямо противоположными точками зрения: от полного «слепого» доверия результатам моделирования до такого же полного пренебрежения ими. Оба случая — результат недостаточной информированности о возможностях используемой CAM ЛП. Ясно, что для успешного применения CAM ЛП виртуальная модель технологического процесса должна соответствовать конкретному производственному процессу. Это соответствие обеспечивается на качественном и количественном уровне. Качественное соответствие возможно за счет использования адекватных математических

моделей тех физических процессов, которые необходимо моделировать.

Современное проектирование литейной технологии осуществляется с помощью CAD-систем и включает в себя построение трехмерных (3D) геометрических моделей детали, отливки с литниковой системой, а также литейной оснастки и изготовление по ним чертежной документации. Но для отработки литейной технологии на стадии проектирования без дорогостоящих натурных экспериментов, а также для оптимизирования уже имеющейся технологии - конфигурации литниковой системы, прибылей, температуры и режима заливки и т.д. - необходимо использовать CAE-системы. В настоящее время на многих зарубежных и некоторых российских машиностроительных предприятиях, имеющих литейное производство, используются те или иные CAE-системы компьютерного моделирования литейных процессов (СКМ ЛП), позволяющие рассчитывать заливку, затвердевание и образование дефектов отливок для различных видов литья.

Первым этапом компьютерного анализа ЛП является построение 3D модели отливки и формы. Большинство СКМ ЛП не имеют собственных средств построения 3D моделей, поэтому необходимо приобретать еще и CAD-систему. CAD-системы высшего уровня (Pro/Engineer, Unigraphics, EUCLID3, CATIA и некоторые другие) предоставляют богатые возможности твердотельного и поверхностного моделирования, но слишком дороги. Из более доступных систем среднего уровня стоит особенно отметить SolidWorks, который прост в освоении, предоставлят мощные средства 3D моделирования и имеет множество приложений, а также Solid Edge, IronCAD, CadKey, Mechanical Desktop, Think3, Vellum Solids и другие. На тех предприятиях, где уже имеются CAD-системы, вопрос выбора не возникает - используется то, что есть. Следующим этапом является представление созданной 3D модели в виде, необходимом для расчетов в СКМ ЛП. Существует три математических метода реализации представления геометрии в подобных системах анализа: метод конечных разностей (FDM), метод конечных элементов (FEM) и

метод граничных элементов (BEM). С помощью FDM реализованы многие системы анализа ЛП (MAGMASOFT, AFS, CastCAE, LVMFlow и др.). Это вызвано простотой реализации данного метода, хотя он имеет существенный недостаток, заключающийся в искажении геометрии при ее ступенчатом представлении (например, периметр круга, представленного FDM, равен периметру описанного вокруг него квадрата), а также при большой разностенности отливок могут возникнуть трудности при адекватном описании геометрии, поэтому в универсальных CAE-системах (таких как ANSYS, NASTRAN, ABAQUS, MARC и др.) где конкуренция очень сильна, он практически не находит применения. FEM позволяет описать геометрию с любой степенью точности, поэтому его применение представляется более предпочтительным (им реализованы такие системы, как ProCAST, SIMTEC, PASSAGE/PowerCAST, ПОЛИГОН и др.). Следовательно, для повышения точности моделирования следует выбирать систему, основанную на FEM. BEM является весьма перспективным, но пока он не нашел применения в СКМ ЛП. Все системы, основанные на FDM, имеют собственные генераторы сетки. Большинство систем, использующие FEM, также имеют встроенные генераторы сетки. Но некоторые системы не имеют 3D генератора сетки, в этом случае нужно использовать либо специализированные пре- и постпроцессоры (HyperMesh, FEMAP, GAMBIT), либо встроенные генераторы других FEA-систем (ANSYS, NASTRAN и др.), а затем импортировать модель в СКМ ЛП.

В различных областях науки и техники широко используется метод математического моделирования. Этот метод включает в себя разработку физических и математических моделей, численных методов и программного обеспечения, проведение численного эксперимента с привлечением средств вычислительной техники (его результаты анализируются и используются в практических целях). В технике и технологии преимущества метода математического моделирования очевидны: оптимизация проектирования, сокращение затрат на отработку, повышение качества продукции, уменьшение эксплуатационных расходов и т.д. Математическое моделирование

существенно преобразует также сам характер научных исследований, устанавливая новые формы взаимосвязи между экспериментальными и математическими методами.

Применение математического моделирования в литейной промышленности привело к появлению большого числа программных пакетов, с помощью которых более или менее успешно решаются задачи, с которыми литейщики сталкиваются в повседневной практике.

В основе математической модели литейных процессов лежат уравнения тепломассопереноса: уравнения теплопроводности, Навье-Стокса, диффузии, кинетические уравнения фазовых превращений и т.д. Расчетная область включает не только объем, занимаемый расплавом, но также и формообразующую среду с различными граничными и начальными условиями. Возможен учет цикличности процесса литья (например, литье в кокиль).Программы для моделирования литейных процессов, распространенные в настоящее время, в основном различаются степенью полноты учитываемых при моделировании факторов.

Второе различие связано с методами получения и решения разностных уравнений: уравнения тепломассопереноса могут быть записаны в дифференциальном или интегральном виде.

Метод конечных разностей (МКР) базируется на уравнениях в дифференциальной форме, при этом дифференциальные операторы заменяются конечно-разностными соотношениями различной степени точности. Как правило, они строятся на ортогональных сетках (прямоугольной, цилиндрической и т.д.). Это позволяет факторизовать операторы и свести решение многомерной задачи к последовательности одномерных задач, а значит намного упростить и ускорить решение общей системы уравнений. К недостаткам следует отнести плохую аппроксимацию границ сложных областей, что не слишком принципиально для уравнений теплопроводности, но довольно существенно для уравнений гидродинамики. Метод также плохо работает в случае тонкостенных отливок, когда толщина стенок становится сравнимой с шагом сетки.

Метод конечных элементов (МКЭ) и **метод конечного объема (МКО)** базируются на уравнениях тепломассопереноса в интегральном виде. Область, в которой решаются уравнения, разбивается на элементы, внутри которых строятся аппроксиманты функций на основе системы базисных функций, определенных на элементе. «Проектируя» интегральные уравнения на эти базисы, получают систему разностных уравнений. Эта система значительно сложнее принятой в МКР, ее решение требует больших ресурсов памяти и немалого времени. Преимущество МКЭ - хорошая аппроксимация границы, недостатки - необходимость добротного генератора конечных элементов, сложность уравнений, невозможность факторизации.

Модификации МКО пытаются соединить в себе простоту и факторизацию МКР и хорошую аппроксимацию границ между различными материалами и различными фазами.

Практика показывает, что оптимальный подход состоит не в выборе какого-то одного метода решения, а в использовании комбинации различных методов – это позволяет получить выигрыш в скорости, точности и адекватности получаемых результатов экспериментальным данным.

ЛИТЕРАТУРА

1. *Бахвалов Н.С.* Численные методы. – М.: Наука, 1975.

3. *Бояринов А.И.*, *Кафаров В.В.* Методы оптимизации в химической технологии. – М.: Химия, 1975.

4. *Брановицкая С.В.*, *Медведев Р.Б.*, *Фиалков Ю.Я.* Вычислительная математика в химии и химической технологии. – Киев: Вища школа, 1986.

5. *Волков Е.А.* Численные методы. – М: Наука, 1987.

6. *Демидович Б.П.*, *Марон И.А.* Основы вычислительной математики. – М.: Наука, 1966.

5. *Калиткин Н.Н.* Численные методы. – М.: Наука, 1978.

6. *Мак-Кракен Д.*, *Дорн У.* Численные методы и программирование на ФОРТРАНЕ. – М.: Мир, 1977.

7. *Плис А.И.*, *Сливина Н.А.* Лабораторный практикум по высшей математике. – М.: Высшая школа, 1983.

8. *Самарский А.А.*, *Гулин А.В.* Численные методы. – М.: Наука, 1989.

9. *Турчак Л.И.* Основы численных методов. – М.: Наука, 1987.

10. *Форсайт Дж.*, *Малькольм М.*, *Моулер К.* Машинные методы математических вычислений. – М.: Мир, 1980.

11. *Фурунжиев Р.И.*, *Бабушкин Ф.М.*, *Варавко В.В.* Применение математических методов и ЭВМ. – Минск: Вышейшая школа, 1988.

12. *Шуп Т.* Решение инженерных задач на ЭВМ. – М.: Мир, 1982.

СОДЕРЖАНИЕ

Жаранов Виталий Александрович

МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ ТЕХНОЛОГИЧЕСКИХ ПРОЦЕССОВ

Курс лекций
по одноименной дисциплине
для студентов специальности 1-36 02 01
«Машины и технология литейного производства»
дневной и заочной форм обучения

Подписано к размещению в электронную библиотеку
ГГТУ им. П О. Сухого в качестве электронного
учебно-методического документа 24.12.09.

Рег. № 113Е.
E-mail: ic@gstu.gomel.by
http://www.gstu.gomel.by