Министерство образования и науки Российской Федерации
Московский государственный институт электронной техники
(технический университет)

В.И. Петраков

Автоматизация схемотехнического проектирования ИС

Курс лекций

Утверждено редакционно-издательским советом института

Москва 2010

УДК 658.512.011.56
П30

Рецензенты: докт. техн. наук, проф. *А.Г. Соколов*;
канд. техн. наук *С.В. Макаров*

Петраков В.И.

П30 **Автоматизация схемотехнического проектирования ИС**: курс лекций. - М.: МИЭТ, 2010. - 116 с.: ил.

ISBN 978-5-7256-0609-6

Курс лекций содержит необходимые сведения о средствах, терминологии и возможностях применения аппарата теории электрических цепей и вычислительной математики для автоматизации этапа схемотехнического проектирования интегральных схем (ИС). Рассмотрены следующие темы: методы формирования математических моделей ИС, методы численного решения систем обыкновенных дифференциальных уравнений, методы решения систем нелинейных уравнений, прямые и итерационные методы решения систем линейных уравнений, методы анализа чувствительности, методы параметрической оптимизации.

Для студентов, специализирующихся в области электроники, и изучающих дисциплины, связанные с проектированием интегральных схем.

ISBN 978-5-7256-0609-6

Курс лекций

Петраков *Владимир Иванович*

Автоматизация схемотехнического проектирования ИС

Редактор *Н.А. Кузнецова*. Технический редактор *Е.Н. Романова*. Корректор *Л.Г. Лосякова*. Верстка автора.

Подписано в печать с оригинал-макета 25.12.2010. Формат 60x84 1/16. Печать офсетная. Бумага офсетная. Гарнитура Times New Roman. Усл. печ. л. 6,73. Уч.-изд. л. 5,8. Тираж 150 экз. Заказ 143.

Отпечатано в типографии ИПК МИЭТ.
124498, Москва, Зеленоград, проезд 4806, д. 5, МИЭТ.

Оглавление

Введение

При проектировании любой электронной схемы разработчик обязательно использует результаты, полученные на этапе схемотехнического проектирования. Если обратиться к истории развития систем автоматизированного проектирования (САПР), то именно со схемотехнического моделирования электронных схем началась эпоха автоматизации процесса проектирования. Как во всякой иной области деятельности, методы, алгоритмы и программы моделирования прошли развитие от простейших вариантов используемых решений до чрезвычайно сложных, включающих последние достижения в области вычислительной математики. В значительной степени все это обусловлено двумя факторами:

1) наблюдаемым быстрым развитием технологий изготовления интегральных схем. Это ведет к уменьшению технологических размеров кристалла и, следовательно, с одной стороны, к резкому увеличению числа элементов, которые могут быть размещены на кристалле, а с другой - к повышению их быстродействия. С этим связано быстрое развитие возможностей новейших средств вычислительной техники;

2) выявлением в полупроводниковых структурах новых физических эффектов, которые ранее не наблюдались. Для учета влияния этих эффектов необходимо разрабатывать новые, более сложные модели компонентов. Это ведет к усложнению соответствующих программных модулей для отдельных компонентов и соответственно увеличению времени машинного расчета характеристик ИС. Кроме того, растет воздействие на работу схем паразитных элементов, возникающих в реальных ИС. Выявление (экстракция) паразитных элементов, определение их параметров возможны только после выполнения этапа топологического проектирования. Учет их влияния требует повторного моделирования принципиальных электрических схем, соответствующих реальной топологии. Число элементов экстрагированных схем существенно больше, чем в исходных. Соответственно, размерность их математических моделей больше размерности исходных. Это ведет к увеличению объема памяти, необходимой для хранения информации о схемах, а также существенно увеличивает общее время моделирования схем.

Основное внимание в задачах ускорения процесса моделирования уделяется модификации математических методов, применяемых для формирования математической модели в виде системы линейных алгебраических уравнений и ее решения. Кроме того, становится очевидной актуальность разработки новых методов и алгоритмов формирования математических моделей схем и новых подходов к их решению.

Таким образом, перед разработчиками средств автоматизации проектирования стоит задача удовлетворить постоянно возрастающие требования проектировщиков больших,

сверхбольших, ультрабольших ИС по сокращению времени моделирования предлагаемых схем при обеспечении точности, гарантирующей достоверность моделирования. Используемые в настоящее время модели и методы созданы в результате компромисса между требованиями к точности и вычислительной эффективности. Одной из особенностей схемотехнического проектирования в процессе моделирования является необходимость иметь общие представления о математическом аппарате моделирования. Это может оказать помощь проектировщику в случаях, когда моделирование при заданных условиях не выполняется.

В предлагаемом конспекте лекций содержится материал о средствах, терминологии и возможностях применения аппаратов теории электрических цепей и вычислительной математики для автоматизации этапа схемотехнического проектирования интегральных схем.

Лекция 1

Общие сведения о дисциплине

Заучивайте на дому
Текст лекции по руководству.
Учитель, сохраняя сходство,
Весь курс читает по нему.

И.В. Гёте. Фауст

Лавинообразное развитие средств вычислительной техники, быстрый рост средств телекоммуникаций, потребительской и автомобильной электроники и т.п. привели к существенному расширению функций, выполняемых интегральными схемами. Соответственно расширилась номенклатура разрабатываемых интегральных схем. В настоящее время значительную долю проектируемых схем составляют схемы, обрабатывающие аналоговые сигналы. Отчасти это связано с широким распространением методологии проектирования систем на кристалле (System on Chip - SoC) на основе библиотек сложно-функциональных блоков (IP-block), бóльшая часть которых является аналого-цифровой либо цифроаналоговой.

Переход на новые технологии, ведущий к уменьшению размеров кристалла и повышению быстродействия, также привел к необходимости рассматривать поведение "цифровых" элементов как "аналоговых", т.е. цифровые сигналы считать аналоговыми и выполнять моделирование цифровых схем как аналоговых на схемотехническом уровне.

При переходе полупроводниковой технологии в нанометровую область в полупроводниковых структурах проявилось множество новых физических эффектов, которые ранее не наблюдались. Для учета влияния этих эффектов необходимы новые модели компонентов. Их появление требует решения задач идентификации параметров компонентов, обоснования достоверности и точности моделирования, разработки новых маршрутов моделирования и их стандартизации. Чрезвычайно важной при этом является задача обучения проектировщиков эффективному использованию современных средств САПР, таких как CADENCE, SYNOPSYS, MENTOR GRAPHICS.

Наряду с отмеченными проблемами существует проблема быстродействия средств моделирования, которая приводит к желанию использовать предельно упрощенные модели компонентов (макромодели) и приближенные методы моделирования электронных цепей. Упрощения неизбежно приводят к уменьшению достоверности моделирования и воз-

растанию неопределенности в области их допустимого применения. Степень неопределенности возрастает также при изменении технологического процесса изготовления сверхбольших интегральных схем (СБИС). Требования к точности наилучшим образом удовлетворяются при использовании электрических либо физико-технологических моделей. Вычислительные затраты на физико-технологическое моделирование очень велики, поэтому традиционный подход к моделированию основан на построении "точных" электрических моделей компонентов принципиальных электрических схем, "точных" моделей самих схем, их математических моделей, представленных в виде элементарных алгебраических функций, систем нелинейных уравнений, систем обыкновенных дифференциальных уравнений и решении полученных уравнений численными методами.

Для решения перечисленных выше проблем созданы такие организации, как совет по компонентным (компактным) моделям (Compact Model Council - CMC), рабочая группа Американского национального института стандартов (NIST Working Group on Model Validation), подкомитет по моделированию при Ассоциации полупроводниковых компаний (FSA Modeling Subcommitte).

Целью данного курса является знакомство с методами и алгоритмами, на основе которых разработаны современные программы схемотехнического проектирования ИС, а также поддержка определенного уровня знаний языков программирования.

К сожалению, развитие технологии изготовления микросхем в последние годы опережает развитие средств моделирования. Поэтому следует уделять особое внимание вопросам автоматизации проектирования в целом и, в том числе, этапу схемотехнического проектирования.

Лекция 2

Проектные процедуры этапа схемотехнического проектирования

Ах, господи, но жизнь-то недолга,
А путь к познанью дальний. Страшно вчуже:
И так уж ваш покорнейший слуга
Пыхтит от рвенья, а не стало б хуже!

И.В. Гёте. Фауст

На каждом этапе проектирования разработчики схем выполняют одни и те же проектные процедуры: структурный синтез, составление математической модели и задание параметров, анализ математической модели, оптимизацию математической модели и ее статистический анализ.

При выполнении схемотехнического проектирования необходимо выполнить следующие основные проектные процедуры:

1) синтез структуры (составление принципиальной электрической схемы;

2) составление математической модели;

3) синтез параметров;

4) анализ работы принципиальной электрической схемы (одновариантный);

5) оптимизацию (структурную и параметрическую, т.е. многовариантный анализ);

6) статистический анализ и оптимизацию (многовариантный анализ).

В этом случае процесс схемотехнического проектирования представим в виде упрощенной блок-схемы (рис.2.1).

Проектирование схемы обычно начинается с составления технического задания и формулирования технических требований на системном уровне. После проверки реализуемости технических требований выполняется функциональный синтез системы и определяются функциональные взаимосвязи между ее регистрами или аналоговыми блоками.

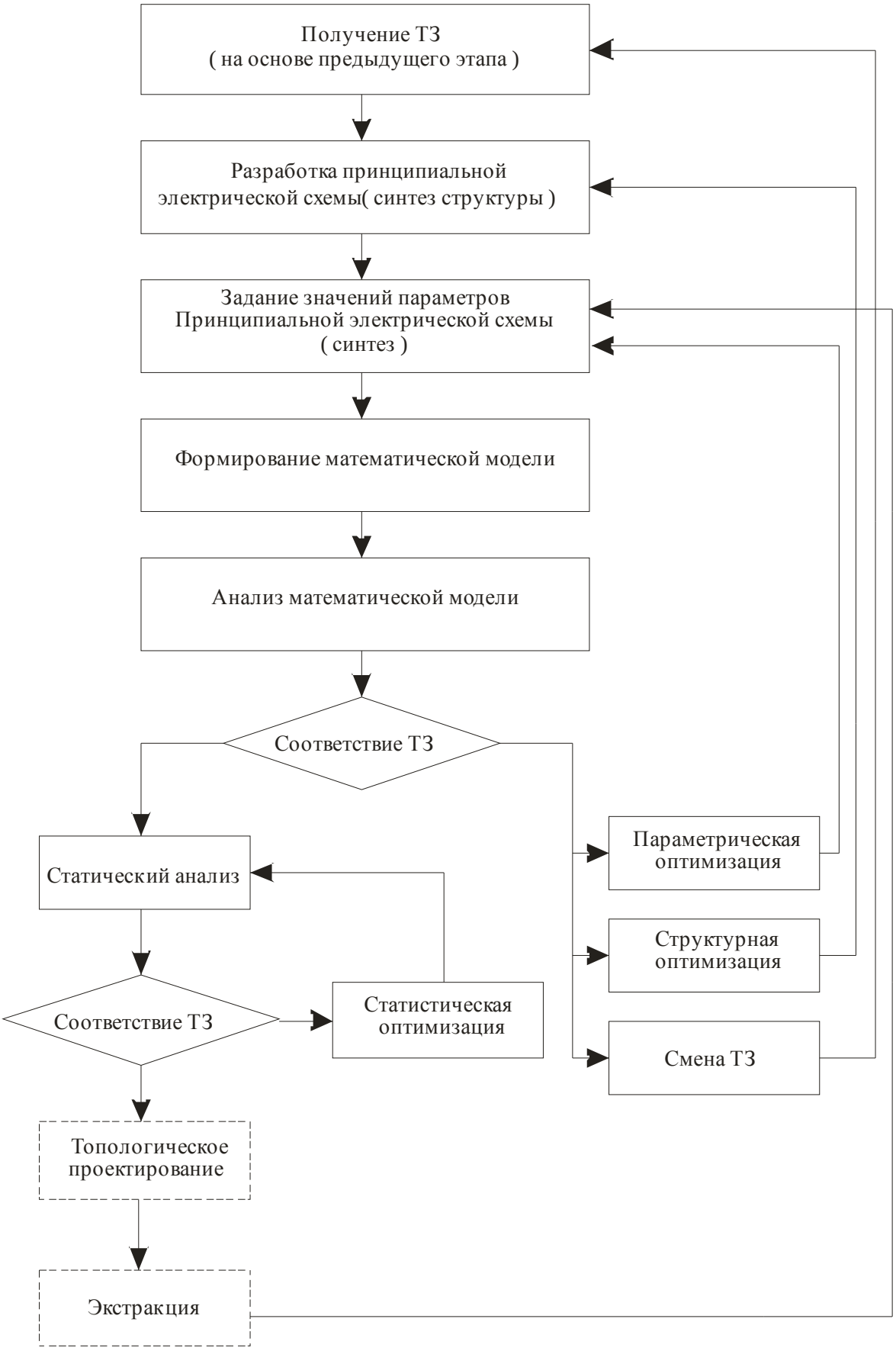


Рис.2.1. Блок-схема процесса схемотехнического проектирования

Проектирование на этом этапе выполняется так, чтобы обеспечить тестируемость изделия после его изготовления. Далее проводится разработка электрических схем, их оптимизация, верификация и синтез топологии СБИС (размещение на кристалле и трассировка).

Схемотехническое моделирование выполняется в два этапа: до проектирования топологии и после него. Повторное моделирование выполняется после применения программ восстановления (экстракции) принципиальных электрических схем из топологии кристалла. Программы экстракции поставляются в комплекте с программами схемотехнического моделирования. Принципиальная электрическая схема, соответствующая разработанной топологии, содержит паразитные элементы, возникающие в реальных физических структурах. В зависимости от сложности проекта циклы схемотехнического моделирования и проектирования топологии могут выполняться на разных уровнях иерархии проекта, чередуясь с этапами верификации топологии и коррекции электрической схемы.

На этапе схемотехнического моделирования следует также учитывать технологический разброс параметров компонентов СБИС, т.е. выполнять статистический анализ, применяя известные методы, такие как метод наихудшего случая либо метод Монте-Карло.

2.1. Основные виды схемотехнического анализа ИС

Понятие анализа включает выбор метода решения математической модели электрической схемы, алгоритма и его программной реализации. Результаты анализа дают ответ на вопрос, какими свойствами обладает объект проектирования, насколько хорошо он удовлетворяет требованиям ТЗ. Программы одновариантного анализа являются основой пакета прикладных программ любого назначения, поскольку последующие проектные процедуры сводятся к многократному решению задачи одновариантного анализа.

2.1.1. Статический анализ (анализ по постоянному току)

При анализе статических характеристик схемы рассчитываются электрические характеристики элементов принципиальной схемы (токи и напряжения на них, потенциалы узлов схемы) в отсутствие изменения входных сигналов, т.е. определяется так называемая рабочая точка схемы, моделируются вольт-амперные характеристики (параметры) схемы. К ним относятся функциональные (потребляемая мощность, нагрузочная способность, помехоустойчивость и т.п.) и тестовые (входные и выходные токи и напряжения и т.п.) параметры схемы.

2.1.2. Анализ переходных процессов

Анализ переходных процессов выполняется в целях определения задержек прохождения импульсных либо гармонических сигналов заданной формы и с заданными параметрами по цепям схемы, а также для определения искажения формы этих сигналов. Как правило, входные сигналы имеют трапецеидальную либо синусоидальную форму.

При анализе следует учитывать, что принципиальные электрические схемы содержат помимо функционально необходимых компонентов большое число компонентов, отражающих паразитные связи. Эти связи, как было отмечено ранее, выявляются в результате экстракции паразитных элементов схемы, выполняемой вслед за этапом топологического проектирования.

2.1.3. Частотный анализ

Этот вид анализа выполняется в целях получения амплитудно-частотных характеристик (АЧХ) и фазо-частотных характеристик (ФЧХ). Он применяется главным образом при моделировании радиоприемных и радиопередающих схем.

Перечисленные виды анализа относятся к классу одновариантных.

2.1.4. Параметрическая оптимизация

Параметрическая оптимизация электронной схемы заключается в определении совокупности электрических параметров компонентов, при которых выходные параметры схемы удовлетворяют требованиям ТЗ, а один из них или несколько с соответствующими коэффициентами (весами) принимают экстремальное значение. Оптимизация является многовариантным видом анализа.

Значительное место при анализе ИС занимает статистический расчет схем.

Статистический расчет ИС - это расчет вероятности того, что вектор выходных параметров *X*, заданный параметрами $\{x_1, x_2, x_3, ..., x_n\}$, находится в области работоспособности схемы. В основе статистического расчета лежит учет влияния технологического разброса параметров компонентов схем на выходные параметры.

При статистическом расчете ИС необходимо провести:

- статистическую обработку результатов измерений или расчет разброса параметров математических моделей компонентов;
- статистический анализ схемы;
- статистическую оптимизацию по параметрам компонентов;
- статистическую оптимизацию по тестовым нормам.

Статистический анализ является многовариантным. Он основан на использовании результатов одновариантного анализа того, либо другого вида.

2.2. Особенности статистического анализа ИС

Процедура статистического расчета заканчивается выдачей документации, которая содержит принципиальную электрическую схему, номиналы и типы компонентов, статические и динамические характеристики, рекомендации по разработке топологии ИС и браковочные нормы на электрические измеряемые параметры ИС для контрольно-измерительного оборудования.

Еще раз отметим тот факт, что принципиальные электрические схемы содержат, помимо функционально необходимых компонентов, большое число компонентов, отражающих паразитные связи. Последние, как известно, выявляются в результате экстракции паразитных элементов схемы, выполняемой вслед за этапом топологического проектирования. Этот факт необходимо учитывать, если ставится задача повышения точности схемотехнического моделирования.

Среди перечисленных ранее проектных процедур наибольшие трудности для автоматизации представляет процедура синтеза структур аналоговых цепей - разработка принципиальных электрических схем. В настоящее время в широко распространенных системах сквозного проектирования БИС, таких как CADENCE, SYNOPSYS, MENTOR GRAPHICS, на этапе схемотехнического проектирования автоматизированы процедуры формирования математических моделей схем (ММС), различных видов одновариантного анализа, параметрической оптимизации, статистического анализа.

Лекция 3

Математические модели электрических схем.

Общие сведения

Во всем подслушать жизнь стремясь,
Спешат явленья обездушить,
Забыв, что если в них нарушить
Одушевляющую связь,
То больше нечего и слушать.

И.В. Гёте. Фауст

Математическая модель схемы (далее будем использовать сокращение ММС) - это совокупность объектов в виде чисел, векторов и связей между ними, которая отражает существенные с точки зрения проектировщика свойства изучаемого объекта (логические, принципиальные, топологические схемы).

На каждом этапе проектирования различают ММС объектов и систем. Математическая модель (ММ) системы, получаемая непосредственно объединением ММ компонентов в общую систему уравнений, называется ***полной ММ***. Математическая модель, более простая по сравнению с полной, и отражающая только наиболее важные с точки зрения проектировщика характеристики, называется ***макромоделью***. При замене полной ММ макромоделью сокращается время моделирования и снижаются требования к оперативной памяти.

Основные требования к ММ:

1) адекватность (точность);

2) простота, позволяющая выполнить расчет за приемлемое время (легкий, эффективный подсчет).

Математической моделью электронной схемы на этапе схемотехнического проектирования для последующего анализа является система уравнений, связывающая токи и напряжения (потенциалы) в различных компонентах схемы. Математическая модель схемы получается объединением математических моделей отдельных компонентов в общую систему уравнений. Результирующую систему получают, применяя методы, развитые в теории электрических цепей. Такая модель называется *алгоритмической*, поскольку она связана с возможными методами и алгоритмами ее решения. Если удается получить модель в виде явной зависимости выходных параметров от внутренних и внешних, то модель называется *аналитической*.

Рассматриваемые далее методы формирования ММС основаны на топологических уравнениях для электрических схем и математических моделях компонентов, входящих в схемы. ***Топологические уравнения*** отражают связи между компонентами в анализируемой схеме. Удобным средством описания этих связей являются графы. Получающиеся при этом топологические структуры представляют собой линейные связные направленные графы. Для получения моделей используются уравнения законов Кирхгофа.

3.1. Определения

Приведем основные определения теории графов.

1) ***Ветвь*** - линейный сегмент, представляющий схемный элемент. В ряде случаев ветвь трактуется как грань графа.

2) ***Узел*** - место соединения ветвей.

3) ***Подграф*** - подмножество ветвей и узлов графа.

Подграф называется правильным, если число узлов и ветвей данного подграфа не превышает числа узлов и ветвей исходного графа.

4) ***Связный граф*** - граф, в котором между любыми двумя узлами существует, по крайней мере, один путь (или одна ветвь).

5) ***Направленный граф*** - граф, ветви которого имеют направления.

6) ***Контур*** - часть графа, образующая единственным образом замкнутый путь.

7) ***Фундаментальным деревом*** графа называют подграф из $\beta - 1$ ветвей, в котором нет замкнутых контуров (циклов). Здесь β - количество узлов. Для любого графа (если сам граф не является деревом) можно построить множество фундаментальных деревьев.

8) ***Хорда*** - ветвь графа, не входящая в дерево.

9) ***Контуром i-й хорды*** называют совокупность ветвей, входящих в замкнутый контур, образуемый при подключении *i*-й хорды к дереву.

10) ***Нормальное дерево графа схемы***. Под нормальным деревом понимается фундаментальное дерево, в котором включение ветвей происходит со следующими приоритетами: ветви источников напряжения, емкостные, резистивные, индуктивные и источников тока.

Топологический граф может быть описан матрицами инциденций, контуров и сечений.

11) ***Матрица инциденций***, или матрица "узел-ветвь". Строки матрицы инциденций соответствуют узлам схемы, а столбцы - ветвям.

В столбце *i*-й ветви записываются единицы на пересечении со строками инцидентных узлов, причем плюс 1 соответствует узлу, в который ток ветви втекает, и минус 1 - узлу, из которого этот ток вытекает. Если ветвь не присоединена к узлу, то элемент матрицы - нуль.

12) ***Матрица контуров и сечений***. Строки матрицы соответствуют хордам, а столбцы - ветвям дерева. В строке *i*-й хорды записываются единицы в тех столбцах, которые соответствуют ветви дерева, входящей в контур *i*-й хорды. Единица берется со знаком плюс, если выбранные направления токов в данной ветви и *i*-хорде совпадают, в противном случае - со знаком минус. Остальные элементы *i*-й строки - нули.

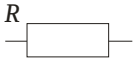
Компонентные уравнения описывают электрические свойства компонентов, из которых составляются принципиальные электрические схемы. Выделяют ***базовые библиотечные компоненты*** электрических схем, а также ***типовые библиотечные компоненты*** и фрагменты схем.

К базовым компонентам относят резисторы, конденсаторы, индуктивности, независимые источники тока и напряжения, а также источники тока и напряжения, управляемые либо током, либо напряжением. Они представимы двухполюсниками. Остальные элементы являются в общем случае многополюсниками. Последние представляются, как правило, в форме эквивалентных схем, состоящих из простых двухполюсных элементов.

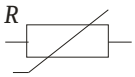
3.1.1. Математические модели (компонентные уравнения)

Приведем общий вид математических моделей для резистора, конденсатора и индуктивности.

Модели для резистора *R*:



$I_R = YU_R$ - линейная модель;



$I_R = Y(U_R)U_R$ - нелинейная модель.

Модели для конденсатора:

$I_C = C\dfrac{dU_C}{dt}$ - линейная модель;

$I_C = C(U)\dfrac{dU_C}{dt}$ - нелинейная модель.

Модели для индуктивности:

$$U_L = L\frac{dI_L}{dt}$$ - линейная модель;

$$U_L = L(U)\frac{dI_L}{dt}$$ - нелинейная модель.

Здесь R, L, C - сопротивление, емкость и индуктивность для линейных моделей компонентов; $R(U)$, $L(U)$, $C(U)$ - для нелинейных.

3.2. Источники

3.2.1. Источники напряжения и тока

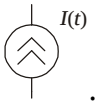
Все источники можно классифицировать по признаку способа получения сигнала. Это - источники независимые и управляемые. Их наиболее абстрактное представление - ветвь графа схемы.

Независимые источники подразделяют на постоянные (не зависящие от времени) и источники, параметры которых изменяются во времени по некоторому закону. Наиболее часто используются импульсные источники, форма сигналов которых описывается трапецией, источники, сигналы которых имеют сложную кусочно-линейную форму, источники гармонических сигналов. В дальнейшем будем обозначать:

1) источник напряжения $E(t)$. Условное обозначение:

.

2) источник тока $I(t)$. Условное обозначение:



.

Управляемые источники - это источники, сигналы которых задаются функцией от тока или напряжения другой ветви графа. Всего выделяют четыре источника:

1) источник напряжения, управляемый либо током $E_m = f(I_n)$, либо напряжением $E_m = f(U_n)$;

2) источник тока, управляемый либо током $I_m = f(I_n)$, либо напряжением $I_m = f(U_n)$.

Здесь m, n - обозначения управляемых и управляющих ветвей соответственно.

Примечание. Как было отмечено ранее, большинство компонентов схем (например, транзисторы) имеют схемные и математические модели, построенные из базовых компонентов и математических моделей базовых компонентов. Подавляющее число компонентов имеют сложные нелинейные вольт-амперные характеристики (ВАХ). При создании моделей активных или пассивных нелинейных компонентов, а также при их использовании необходимо искать компромисс между требованиями к точности моделирования и временем моделирования. В настоящее время в связи с уменьшением допустимых технологических размеров и, соответственно, быстрым ростом производительности компьютеров на первое место выдвигают требования к достоверности моделирования.

3.2.2. Примеры схемотехнических и математических моделей некоторых компонентов

Схемотехническая модель диода (*p* - *n*-перехода). Схемотехническую модель полупроводникового диода можно представить в виде эквивалентной схемы, составленной из базовых библиотечных элементов (рис.3.1).

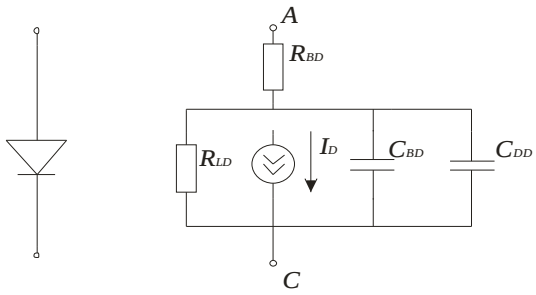


Рис.3.1. Схемотехническая модель полупроводникового диода

Из приведенной схемы видно, что модель диода состоит из пяти компонентов:

1) R_{BD} - объемное сопротивление *p*- и *n*-областей плюс сопротивление контактов выводов;

2) R_{LD} - сопротивление утечки перехода;

3) C_{BD} - барьерная емкость *p* - *n*-перехода;

4) C_{DD} - диффузионная емкость *p* - *n*-перехода;

5) $I_D = I_{D0}(e^{\frac{U_D}{m\varphi_T}} - 1)$ - ток диода при прямом и обратном смещениях.

Здесь m - показатель неидеальности *p* - *n*-перехода; $\varphi_T = \frac{kT}{q}$, где k - постоянная Больцмана; T - температура в градусах Кельвина; q - заряд электрона.

Каждый из перечисленных параметров описывается нелинейными уравнениями.

Основное уравнение для тока диода имеет вид

$$I = I_D + (C_{BD} + C_{DD})\frac{d}{dt}\cdot U_D + \frac{1}{R_{LD}}\cdot U_D, \qquad (3.1)$$

где U_D - напряжение на p - n-переходе.

Пренебрегая влиянием первых четырех компонентов схемной модели, можно существенно упростить формулу (3.1). Такое упрощение позволяет быстро оценить саму возможность разрабатываемой схемы выполнять требуемую функцию.

В ряде случаев, например для начального "ручного" выбора параметров компонентов схемы, желательны упрощенные модели для работы в узком заранее заданном диапазоне изменения напряжений и токов. При этом "точные" нелинейные модели целесообразно заменять линеаризованными.

Например, нелинейную зави-симость $I_D = I_{D0}(e^{\frac{U_D}{m\varphi_T}} - 1)$ вышеприведенной нелинейной модели диода можно линеаризовать следующим образом (рис.3.2).

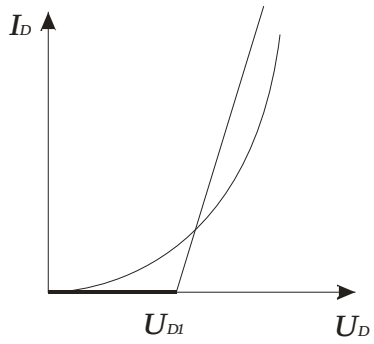


Рис.3.2. Вольт-амперная характеристика диода - нелинейная и линеаризованная (кусочно-линейная)

Из рис.3.2 видно, что для линеаризованной модели $I_D = 0$ на промежутке $U_D = 0 \div U_{D1}$) и имеет линейную зависимость $I_D = g(U_D - U_{D1})$ при $U_D > U_{D1}$.

3.2.3. Упрощенная схемотехническая и математическая модель МДП-транзистора

На рис.3.3 приведена упрощенная схемотехническая модель транзистора.

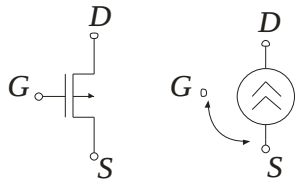


Рис.3.3. Упрощенная схемотехническая модель МДП-транзистора

Предлагаемая модель представляет собой источник тока, управляемый напряжением. Здесь *G* - затвор, *S* - исток, *D* - сток. Математическая модель, соответствующая данной для схемы включения транзистора "общий исток", может быть представлена формулой

$$I_D = F(U_G), \qquad (3.2)$$

где I_D - ток стока; U_G - напряжение затвор-исток.

Обычно на статических вольт-амперных характеристиках выделяют два участка - крутой и пологий.

Для крутого участка ВАХ

$$I_D = g\left[(U_{GS} - U_0)U_{DS} - \frac{1}{2}U_{DS}^2\right]. \qquad (3.3)$$

Для пологого участка ВАХ

$$I_D = \frac{1}{2}g[U_{GS} - U_0]. \qquad (3.4)$$

Упрощенная схемная модель биполярного транзистора приведена на рис.3.4.

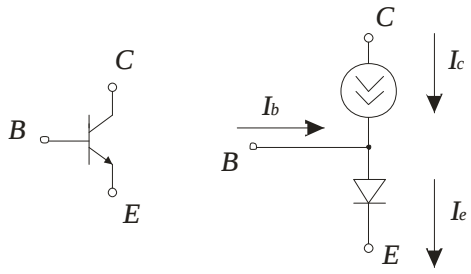


Рис.3.4. Условное обозначение транзистора
и простейшая схемная модель

Предлагаемая модель включает источник тока, управляемый током, а также модель нелинейного резистора (*p* - *n*-перехода).

Здесь I_C - ток коллектора, I_B - ток базы, I_E - ток эмиттера, β - коэффициент передачи тока базы.

$$I_C = \beta \cdot I_B . \quad (3.5)$$

Приведенные примеры моделей элементов показывают, что ММС на этапе схемотехнического проектирования могут быть системами обыкновенных дифференциальных уравнений вида $f(x, \frac{dx}{dt}, t) = 0$, либо $\frac{dx}{dt} = f(x, t)$ (форма Коши), системами нелинейных уравнений вида $f(x, t) = 0$, системами линейных алгебраических уравнений $A \cdot x + b = 0$.

Лекция 4

Методы формирования математических моделей

Иной на то полжизни тратит,
Чтоб до источников дойти,
Глядишь - его на полпути
Удар от прилежанья хватит.

И.В. Гёте. Фауст

Как было отмечено ранее, в основе методов формирования математических моделей лежат законы Кирхгофа. Для каждого метода формирования ММ характерны свои правила выбора системы исходных топологических уравнений и базиса независимых переменных.

В зависимости от этого выбора полученные математические модели - системы уравнений - могут отличаться по виду и по размерности. В наиболее общем случае они представляют собой системы интегро-дифференциальных уравнений. Такие модели целесообразно использовать в открытых системах моделирования, допускающих применение нескольких методов решения. В процессе решения полученные аналитические модели подвергаются преобразованиям путем алгебраизации. В результате получаются либо системы конечно-разностных уравнений, либо алгебраических выражений. Алгебраизированные системы исходных уравнений, в свою очередь, подвергаются линеаризации. Решение таких систем уравнений непосредственно реализуется в виде алгоритмов и программ.

Таким образом, ММС можно классифицировать по виду получающихся выражений, выделив пять классов: системы нелинейных интегро-дифференциальных уравнений, системы нелинейных дифференциальных уравнений, системы нелинейных алгебраических уравнений, системы линейных алгебраических уравнений, системы алгебраических выражений.

Рассмотрим некоторые методы формирования ММС.

Наиболее известными методами являются: 1) табличный метод; 2) метод переменных состояния; 3) метод узловых потенциалов; 4) модифицированный метод узловых потенциалов.

4.1. Табличный метод

Табличный метод представляет собой систему исходных топологических и компонентных уравнений, не подвергшихся никаким преобразованиям. В вектор базисных координат включаются токи и напряжения всех ветвей схемы (за исключением величин, зависящих только от времени или постоянных). Исходными топологическими уравнениями являются уравнения законов Кирхгофа.

$$U_X + M \cdot U_{\text{вд}} = 0;\ \ I_{\text{вд}} - M^t \cdot I_X = 0, \quad (4.1)$$

где U_X и I_X - напряжения и токи ветвей, являющихся хордами; $U_{\text{вд}}$ и $I_{\text{вд}}$ - напряжения и токи ветвей дерева; M - топологическая матрица контуров и сечений.

Разделение ветвей на хорды и ветви дерева определяется выбором фундаментального дерева в графе схемы.

В качестве примера, иллюстрирующего применение табличного метода, используем эквивалентную схему (рис.4.1), для которой на рис.4.2 приведен соответствующий граф.

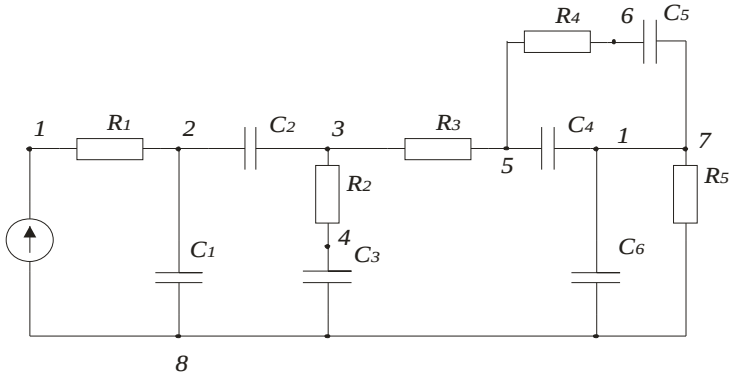


Рис.4.1. Схема электрическая принципиальная

Для графа, изображенного на рис.4.2, одно из возможных фундаментальных деревьев - множество ветвей дерева ВД:

ВД = {E_1, C_1, C_2, C_3, C_4, C_5, C_6} и хорд ВХ:

ВХ = {R_1, R_2, R_3, R_4, R_5}.

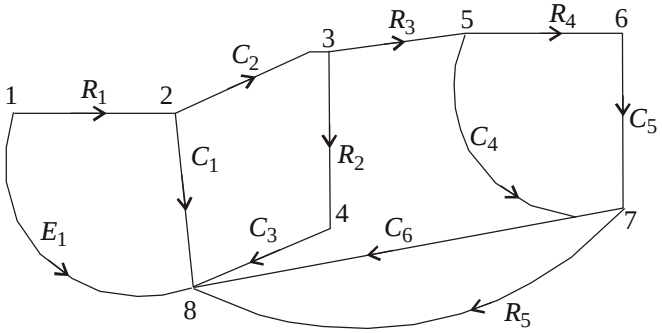


Рис.4.2. Граф эквивалентной схемы рис.4.1

Граф состоит из тех же ветвей (ребер) и узлов (вершин), что и эквивалентная схема, и отличается от схемы отсутствием условных изображений двухполюсников в ветвях.

Матрица *M* для данного примера имеет следующий вид:

	E_1	C_1	C_2	C_3	C_4	C_5	C_6
R_1	-1	$+1$	0	0	0	0	0
R_2	0	-1	$+1$	$+1$	0	0	0
R_3	0	-1	$+1$	0	$+1$	0	$+1$
R_4	0	0	0	0	-1	$+1$	0
R_5	0	0	0	0	0	0	-1

От выбора фундаментального дерева зависит насыщенность матрицы *M*. В алгоритмах табличного метода используются такие правила построения дерева, которые приводят к минимальной насыщенности матрицы контуров и сечений. Недостатком метода является то, что полученная ММС имеет сравнительно высокий порядок.

Напомним, что в практике применения метода исходные уравнения подвергаются алгебраизации и линеаризации.

4.2. Метод переменных состояний

Метод переменных состояний предназначен для получения ММС как системы обыкновенных дифференциальных уравнений (ОДУ) в форме Коши. Базисными переменными в этом методе являются так называемые переменные состояния, т.е. фазовые переменные, непосредственно характеризующие запас энергии в элементах электрической схемы. К таким переменным относят независимые друг от друга напряжения на емкостях и токи через индуктивности. Исходные топологические уравнения те же, что в табличном методе. От-

личия заключаются в том, что топологическая матрица контуров и сечений M получается на основе построения ***нормального*** дерева графа (см. лекцию 3).

При преобразованиях компонентных уравнений стремятся получить уравнения, выражающие емкостные токи I_C и напряжения на индуктивностях U_L через переменные состояния. Далее, заменяя I_C и U_L производными переменных состояния, получают ММС.

В рассмотренном примере было выбрано нормальное дерево графа.

Топологические уравнения в развернутом виде представляются следующим образом:

$$\begin{cases} U_{R_1} = U_{E_1} - U_{C_1}; \\ U_{R_2} = U_{C_1} - U_{C_2} - U_{C_3}; \\ U_{R_3} = U_{C_1} - U_{C_2} - U_{C_4} - U_{C_6}; \\ U_{R_4} = U_{C_4} - U_{C_5}; \\ U_{R_5} = U_{C_6}. \end{cases} \qquad (4.2)$$

$$\begin{cases} I_{E_1} = -I_{R_1}; \\ I_{C_1} = I_{R_1} - I_{R_2} - I_{R_3}; \\ I_{C_2} = I_{R_2} + I_{R_3}; \\ I_{C_3} = I_{R_2}; \\ I_{C_4} = I_{R_3} - I_{R_4}; \\ I_{R_5} = I_{R_4}; \\ I_{C_6} = I_{R_3} - I_{R_5}. \end{cases} \qquad (4.3)$$

В системах уравнений (4.2), (4.3) индекс при переменных означает принадлежность фазовой переменной к конкретной ветви. Для получения системы ОДУ в нормальной форме Коши следует в левых частях уравнения (4.3) заменить I_{C_j} на $C_j \dfrac{dU_{C_j}}{dt}$, а в правых частях вместо I_{R_i} подставить $\dfrac{U_{R_i}}{R_i}$ из (4.2).

Форма Коши получается, если в схеме нет ***топологических вырождений*** - наличия замкнутых контуров из ветвей, состоящих из емкостей и источников напряжения или сечений, включающих только индуктивные ветви и ветви источников тока.

В противном случае для расчета вектора $\dfrac{dU}{dt}$ приходится вводить процедуру решения системы линейных алгебраических уравнений, либо устранять вырождения, включая в емкостные контуры и индуктивные сечения дополнительные резистивные ветви, сопро-

тивления которых подбирают так, чтобы можно было пренебречь вносимыми ими погрешностями.

Форма Коши предпочтительна при применении в дальнейшем явных методов численного интегрирования. В этом случае исходные компоненты уравнения не требуется предварительно алгебраизовать и линеаризовать. Решение на каждом шаге интегрирования представляет собой набор алгебраических функций, аргументы которых являются константами.

4.3. Метод узловых потенциалов

Метод узловых потенциалов (МУП) получил широкое распространение в современных программах схемотехнического анализа электронных схем. Исходные топологические уравнения - это уравнения закона токов Кирхгофа, записанные для узлов схемы. Матричная форма записи имеет вид

$$AI(\varphi) = 0, \quad (4.4)$$

где A - матрица инциденций; $I(\varphi)$ - вектор токов ветвей.

В данном методе в качестве независимых переменных выбираются потенциалы φ_n узлов цепи относительно некоторого базисного опорного узла. Как правило, в качестве опорного (нулевого) узла выбирается "земляной" узел. Из общего вида уравнений (4.4) следует, что МУП применим для схем, включающих ветви, имеющие конечное сопротивление, а также ветви, токи которых управляются напряжениями других ветвей. Это элементы следующих типов: пассивные *R, C, L* элементы (линейные либо управляемые напряжениями), независимые источники тока *I*(*t*), источники тока, управляемые напряжениями $I_m = f(U_n)$.

Если в состав исходной схемы входит источник напряжения, то при получении математической модели его следует заменить на эквивалентный источник тока. Для этого необходимо, чтобы источник напряжения имел конечное внутреннее сопротивление, т.е. был неидеальным. Желательно, чтобы модели индуктивности и конденсатора также включали омические сопротивления. Это позволяет применять одинаковые топологические уравнения, как для временно́го, так и для статического анализа.

Метод узловых потенциалов в общем случае приводит к системе обыкновенных дифференциальных уравнений, неразрешенных относительно производных, т.е. к неявной форме ОДУ. В общем виде система записывается следующим образом:

$$f(x, \frac{dx}{dt}, t) = 0. \qquad (4.5)$$

4.3.1. Пример формирования ММС МУП

Составим математическую модель для принципиальной электрической схемы (рис.4.3).

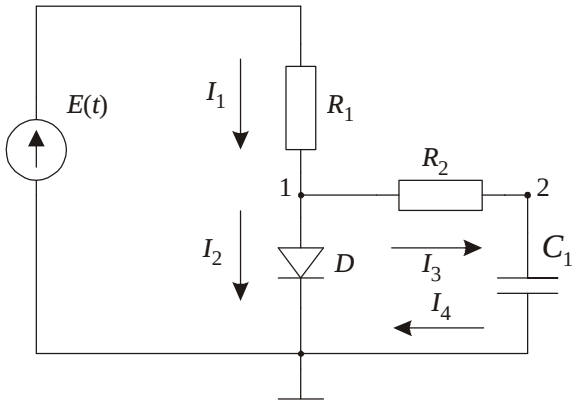


Рис.4.3. Схема принципиальная электрическая

Составим систему уравнений по МУП для узлов 1 и 2:

$$\begin{cases} I_1 - I_2 - I_3 = 0; \\ I_3 - I_4 = 0. \end{cases} \qquad (4.6)$$

Токи ветвей описываются следующим образом:

$$I_1 = \frac{E(t) - \varphi_1}{R_1}; \; I_2 = I_0(e^{\frac{\varphi_1}{\varphi_T}} - 1); \; I_3 = \frac{\varphi_1 - \varphi_2}{R_2}; \; I_4 = C\frac{d\varphi_2}{dt}.$$

Подставив эти выражения в систему (4.6), получим:

$$\begin{cases} \frac{E(t) - \varphi_1}{R_1} - I_0(e^{\frac{\varphi_1}{\varphi_T}} - 1) - \frac{\varphi_1 - \varphi_2}{R_2} = 0; \\ \frac{\varphi_1 - \varphi_2}{R_2} - C\frac{d\varphi_2}{dt} = 0. \end{cases} \qquad (4.7)$$

4.4. Модифицированный метод узловых потенциалов

Модифицированный метод узловых потенциалов представляет собой соединение метода узловых потенциалов с методом линеаризации характеристик нелинейных элементов с помощью итерационных методов решения нелинейных уравнений, например метода Ньютона. Метод применяется по той причине, что обычный метод узловых потенциалов имеет известные ограничения, накладываемые на модели компонентов и соответственно на типы переменных.

В модифицированном методе узловых потенциалов для схем с источниками напряжения, а также с элементами, параметры которых зависят от тока, вводятся дополнительные переменные в виде токов указанных элементов (ветвей) и составляются дополнительные уравнения в виде вольт-амперных связей этих ветвей. Токи некоторых элементов могут рассматриваться как независимые (дополнительные) переменные, так и управляемые. В этом случае они рассматриваются как выходные, так и невыходные (зависимые) переменные.

В матричной форме уравнения модифицированного МУП принимают вид

$$\begin{vmatrix} Y_R & B \\ C & D \end{vmatrix} \cdot \begin{vmatrix} U_0 \\ I \end{vmatrix} = \begin{vmatrix} G \\ F \end{vmatrix}, \qquad (4.8)$$

где Y_R - сокращенная подматрица узловых проводимостей, не учитывающая элементов, управляемых током; B - подматрица частных производных от полученных по закону токов Кирхгофа уравнений по дополнительным переменным; U_0 - вектор потенциалов узлов; G - вектор независимых источников тока; C и D - подматрицы, представляющие вольт-амперные связи, дифференцированные по дополнительным переменным; I - вектор дополнительных переменных (токов), представленный подматрицами; F - вектор, характеризующий вклад реактивных элементов в полный вектор независимых источников тока. При этом реактивные элементы рассматриваем только во временно́й области с учетом их конечно-разностного представления.

4.4.1. Пример формирования ММС-схемы с использованием модифицированного метода узловых потенциалов

Сформируем ММС для схемы, представленной на рис.4.4.

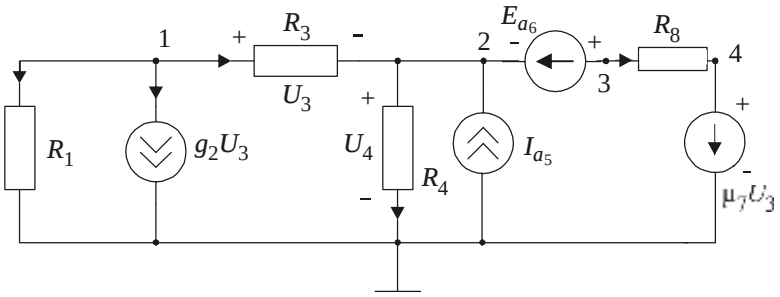


Рис.4.4. Схема электрическая принципиальная

Составим систему уравнений с помощью МУП для узлов 1 - 4:

$$\begin{cases} I_1 + I_2 + I_3 = 0; \\ -I_3 + I_4 - I_5 - I_6 = 0; \\ I_6 + I_8 = 0; \\ I_7 - I_8 = 0. \end{cases} \qquad (4.9)$$

Затем, как и в методе узловых потенциалов, заменяем токи согласно уравнениям ветвей для узлов 1 - 4:

$$\begin{cases} \frac{1}{R_1} \cdot U_1 + g_2 \cdot U_3 + \frac{1}{R_3} \cdot U_3 = 0; \\ -\frac{1}{R_3} \cdot U_3 + \frac{1}{R_4} \cdot U_4 - i_6 = I_{a_5}; \\ i_6 + \frac{1}{R_8} \cdot U_8 = 0; \\ -\frac{1}{R_8} \cdot U_8 + i_7 = 0. \end{cases} \qquad (4.10)$$

Для определения токов i_6 и i_7 к этим четырем уравнениям добавим еще два.

Для ветви 6: $U_6 = E_{a_6}$.

Для ветви 7: $U_7 - \mu_7 U_3 = 0$.

Теперь заменим все напряжения ветвей разностями соответствующих узловых потенциалов.

Для узлов 1 - 4:

$$\frac{1}{R_1} \cdot U_1 + g_2 \cdot (U_1 - U_2) + \frac{1}{R_3} \cdot (U_1 - U_2) = 0;$$

$$-\frac{1}{R_3}\cdot(U_1-U_2)+\frac{1}{R_4}\cdot U_2-i_6=I_{a_5};\qquad(4.11)$$

$$i_6+\frac{1}{R_8}\cdot(U_3-U_4)=0;$$

$$-\frac{1}{R_8}\cdot(U_3-U_4)+i_7=0.$$

Для ветви 6: $U_3-U_2=E_{a_6}$.

Для ветви 7: $U_3-\mu_7\cdot(U_1-U_2)=0$.

В матричной форме полученные уравнения имеют вид

$$\begin{vmatrix} \frac{1}{R_1}+g_2+\frac{1}{R_3} & -g_2-\frac{1}{R_3} & 0 & 0 & 0 & 0 \\ -\frac{1}{R_3} & \frac{1}{R_3}+\frac{1}{R_4} & 0 & 0 & -1 & 0 \\ 0 & 0 & \frac{1}{R_8} & -\frac{1}{R_8} & 1 & 0 \\ 0 & 0 & -\frac{1}{R_8} & \frac{1}{R_8} & 0 & 1 \\ 0 & -1 & 1 & 0 & 0 & 0 \\ \mu_7 & -\mu_7 & 0 & 1 & 0 & 0 \end{vmatrix} \cdot \begin{vmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \\ i_6 \\ i_7 \end{vmatrix} = \begin{vmatrix} 0 \\ I_{a_5} \\ 0 \\ 0 \\ E_{a_6} \\ 0 \end{vmatrix}.$$

В заключение отметим, что ток ветви источника напряжения всегда вводится как дополнительная переменная независимо от вида источника. Это правило сохраняется для индуктивностей. Для источников тока, резисторов и конденсаторов дополнительные переменные вводятся в случаях, когда параметры нелинейных элементов зависят от их токов и когда токи ветвей берутся как выходные.

4.5. Специфика математических моделей БИС

Известно, что математические модели БИС имеют большую размерность, характеризующуюся порядком соответствующей системы уравнений. При представлении в матричной форме не более 10% элементов матрицы оказываются ненулевыми. Такое свойство модели называется ***разреженностью.*** Очевидно, что хранить нулевые элементы таких матриц и выполнять операции с ними нецелесообразно. Поэтому в практике схемотехнического моделирования нашли применение способы хранения только ненулевых элемен-

тов матриц. Некоторые способы хранения матриц с учетом их разреженности приведены в приложении 1.

Элементы схемы (например резисторы) имеют большой разброс номинальных значений параметров. Это ведет к большому разбросу собственных значений матриц параметров элементов схем (например, матрицы узловых проводимостей при применении метода узловых потенциалов). Пусть число обусловленности матрицы

$$\left(\mathrm{cond}A = \frac{\lambda_{\max}}{\lambda_{\min}} \right) > 10^4 .$$

Здесь $\lambda_{\max}$ и $\lambda_{\min}$ - соответственно максимальное и минимальное ***собственные значения***. В этом случае система уравнений является ***жесткой.***

Жесткость уравнений обусловливает трудности их решения. Влияние жесткости на результаты моделирования применительно к различным методам решения тех или иных систем уравнений будет рассмотрено далее.

Лекция 5

Основы динамического анализа электронных схем

Едва я миг отдельный возвеличу,
Вскричав: "Мгновение, повремени!" -
Все кончено, и я твоя добыча,
И мне спасенья нет из западни.

И.В. Гёте. Фауст

Целью динамического анализа является моделирование процессов распространения электрических сигналов в электронных схемах. Математически эти процессы, как было показано в лекции 4, описываются системами обыкновенных дифференциальных уравнений вида $f(x, \frac{dx}{dt}, t) = 0$ (общий вид), либо $x'(t) = f(x,t)$ (форма Коши).

Следовательно, задача моделирования переходных процессов в электронных схемах сводится к задаче интегрирования с начальными условиями, т.е. к задаче Коши.

5.1. Задача Коши

Пусть

$$x'(t) = f(x,t) \quad (5.1)$$

при условии $x(a) = x_0$ при $t \in [a,b]$.

Основное предположение относительно вышеприведенного уравнения состоит в том, что система удовлетворяет условию Липшица $\|f(x_1, t) - f(x_2, t)\| \le L\|x_1 - x_2\|$ в равномерной метрике для всех $t \in [a,b]$ и для всех компонентов векторов. При этом можно доказать единственность решения данной задачи. Таким образом, задача Коши - это задача интегрирования с начальными условиями.

Поскольку решить эту задачу даже для систем уравнений малой размерности аналитически невозможно, прибегают к численным методам. Цель ***численного интегрирования*** - нахождение $x(t)$ для моментов времени $t_1, t_2, t_3, ..., t_k$, где $t_{i+1} = t_i + h_i$ (h_i - шаг интегрирования). Иными словами, построение численных алгоритмов решения основано на дискретизации задачи путем замены непрерывного временно́го интервала интегрирования на дискретный. Для этого делят интервал времени моделирования $[a, b]$ на небольшие при-

ращения (вводят дискретный набор точек t_k). Точки набора называют узлами интегрирования, или узлами сетки. Каждое приращение $h_k = \Delta t_k$ называют шагом интегрирования, или шагом сетки, а совокупность узлов - сеточной областью, или сеткой узлов при одновременной замене производных конечно-разностными алгебраическими выражениями.

В результате исходная система ОДУ заменяется системой ***конечно-разностных*** уравнений. Под конечно-разностными уравнениями понимаются алгебраические соотношения между компонентами, отнесенные к узлам сетки. При этом приближенное значение x_{k+1} вычисляется с учетом значений величин, найденных ранее для предыдущих узлов сетки. Если формула, по которой вычисляется x_{k+1}, зависит явно только от x_k, то метод называется ***одношаговым***. Если x_{k+1} вычисляется по двум предыдущим значениям x_k, x_{k+1}, то метод называется ***двухшаговым***.

Следовательно, методы можно классифицировать по признаку числа предыдущих узлов временно́й сетки, значения переменных в которых используются для вычисления переменных в текущем узле сетки. Применение **многошаговых методов** доставляет определенные трудности, например, на первом шаге интегрирования, когда нет предыдущих узлов. В этом случае на первых шагах интегрирования можно воспользоваться одношаговыми методами.

Поскольку численный метод не позволяет найти точное решение $x(t_k)$, обозначим вычисленное значение для момента времени $t = t_k$ как x_k. Равенство $\varepsilon_k = \|x(t_k) - x_k\|$ называют локальной ошибкой при $t = t_k$. ***Локальная ошибка*** состоит из двух компонентов - методической ошибки и ошибки округления, в предположении, что значение x на предыдущем шаге известно точно. Методическую ошибку называют также ***алгоритмической***, поскольку она зависит от вида численного алгоритма разностной аппроксимации производных. Граница ***методической ошибки*** часто обозначается, как "O", а сама локальная методическая ошибка, как $\varepsilon_м = O(h^{p+1})$ при $h \to 0$. Запись указывает, что локальная методическая ошибка стремится к нулю с такой же скоростью, как и h^{p+1}. При этом говорят, что это "***метод p-го порядка***". Следовательно, методы численного интегрирования можно классифицировать по критерию "***порядок метода p***".

Примечание. Почему не пользуются названием "***метод p + 1-го порядка***"? Дело в том, что, применяя "метод p-го порядка", мы при достаточно малом шаге часто получаем ***глобальную*** ошибку, пропорциональную h^p. О глобальной ошибке речь пойдет далее.

В настоящее время большинство программ не использует постоянный шаг, но понятия "локальная ошибка" и "порядок метода" сохраняют свои значения. Эти понятия показывают следующее.

1. При одинаковом шаге метод более высокого порядка чаще обеспечивает более высокую точность по сравнению с методом меньшего порядка.

2. При построении алгоритмов автоматического выбора шага интегрирования основой критерия выбора величины следующего шага может стать утверждение, что повторное вычисление от момента времени $t = t_k$ с новым шагом h_k изменяет ошибку в следующий момент времени $t = t_{k+1}$ примерно в $\left(\frac{h^{k+1}}{h^k}\right)^p$ раз. Локальная ошибка округления зависит от типа вычислительной машины, т.е. она не может быть уменьшена для данной машины, однако различные методы интегрирования по-разному влияют на ошибку округления. Важно помнить, что общая ошибка округления при $t = t_k$ не равна сумме локальных ошибок округления, возникающих на каждом шаге. Поэтому ***для сравнения точности двух алгоритмов необходимо сравнивать их в одни и те же моменты времени*** t_k ***при одном и том же начальном состоянии.***

Помимо локальной ошибки рассматривается также глобальная ошибка. Она является разностью между вычисленным и "теоретическим" решениями. Эта ошибка состоит из двух частей - локальной ошибки и распространяемой ошибки. Глобальная ошибка в узле t_{k+1} - это глобальная ошибка в узле t_k, умноженная на некоторую величину, называемую ***множителем перехода***, плюс локальная ошибка в узле t_{k+1}. При этом с увеличением числа шагов как методическая, так и ошибка округления могут накапливаться. Метод, обладающий свойством уменьшения ошибки округления при увеличении числа шагов, называется ***численно-устойчивым***. В противном случае он является ***численно-неустойчивым.*** Устойчивость метода, как правило, зависит от выбора шага интегрирования. В этом случае метод относят к классу ***условно-устойчивых***. Если метод устойчив при любом шаге интегрирования, то его относят к классу ***глобально (абсолютно, A-)-устойчивых методов***.

5.2. Устойчивые и неустойчивые уравнения

Будем называть уравнение $f(x,t) = 0$ ***устойчивым***, если кривые семейства решений этого уравнения сходятся по мере удаления времени t от начальной точки. Причиной появления семейства решений является изменение начальных условий.

Если кривые решений расходятся по мере удаления времени t от начальной точки, то уравнение назовем ***неустойчивым***. Устойчивые уравнения подавляют ошибки представления чисел. Системы уравнений обладают такими же свойствами, но их труднее интерпретировать графически. Устойчивость систем, как отмечалось ранее, непосредственно связана с собственными значениями матрицы Якоби исходной системы.

Лекция 6

Методы численного интегрирования

Чего ученый счесть не мог -
То заблужденье и подлог.
И.В. Гёте. Фауст

6.1. Явный и неявный методы Эйлера. Метод трапеций

Среди многочисленных методов интегрирования важное место занимают методы, основанные на представлении интересующей нас функции рядом Тейлора. Если вместо точной формулы Тейлора воспользоваться двумя первыми членами ряда, то получим формулы методов, получивших название методов Эйлера.

Пусть система обыкновенных дифференциальных уравнений имеет вид

$$x'(t) = f(x,t). \qquad (6.1)$$

Представим производную как отношение приращения функции к приращению аргумента на основе известной теоремы о среднем.

Теорема о среднем.

$$\frac{f(b)-f(a)}{b-a} = f'(c), \; a \le c \le b. \qquad (6.2)$$

Определяя производную для левой точки интервала, получим формулу:

$$x_{n+1} = x_n + hf(x_n, t_n). \qquad (6.3)$$

Определяя производную для правой точки интервала, получим формулу:

$$x_{n+1} = x_n + hf(x_{n+1}, t_{n+1}). \qquad (6.4)$$

Формулой (6.3) описывается ***явный метод Эйлера (ЯМЭ)***, а формулой (6.4) - ***неявный метод Эйлера (НЯМЭ).*** Из формулы (6.3) следует, что для получения переменной на очередном шаге интегрирования достаточно вычислить ***алгебраические функции***, аргу-

ментами которых служат значения переменных, полученные на предыдущем шаге. Из формулы (6.4) следует, что применение неявного метода Эйлера приводит к ***системе нелинейных уравнений*** на каждом шаге интегрирования.

Комбинируя явный и неявный методы Эйлера, получим формулу ***метода трапеций***:

$$x_{n+1} = x_n + \frac{1}{2}h(f(x_{n+1}, t_{n+1}) + f(x_n, t_n)). \quad (6.5)$$

6.2. Оценка локальной методической погрешности ЯМЭ и НЯМЭ

Получим формулу локальной методической ошибки для явного метода Эйлера.

Заменив в формуле $x_{n+1} = x_n + hf(x_n, t_n)$ $f(x_n, t_n)$ на производную $x'(t_n)$ (см. формулу (6.1)), получим:

$$x_{n+1} = x_n + hx_n'(t_n). \quad (6.6)$$

Поскольку при вычислении локальной методической ошибки значение переменной на предыдущем шаге считается точным, представим формулу (6.3) в виде

$$x_{n+1} = x(t_n) + hx_n'(t_n). \quad (6.7)$$

Разложим функцию в ряд Тейлора в окрестности точки x_n с точностью до члена второго порядка малости

$$x(t_{n+1}) = x_n + hx'(t_n) + \frac{1}{2}h^2x''(\tau), \quad (6.8)$$

где $t_n < \tau < t_{n+1}$, $h = t_{n+1} - t_n$.

Сравнивая выражения (6.7) и (6.8), получим формулу локальной методической ошибки:

$$\varepsilon_м = \frac{1}{2}h^2x''(\tau). \quad (6.9)$$

Заметим, что локальная методическая ошибка $\varepsilon_м$ довольно велика, поэтому для получения приемлемой точности с помощью явного метода Эйлера необходимо выбирать очень маленькую величину шага.

Аналогично выведем формулу локальной методической ошибки для неявного метода Эйлера. Для этого преобразуем формулу (6.4) к виду

$$x_{n+1} = x_n + hx'(t_{n+1}). \qquad (6.10)$$

Поскольку при вычислении локальной методической ошибки значение переменной на предыдущем шаге считается точным, перейдем к следующей формуле:

$$x_{n+1} = x(t_n) + hx'(t_{n+1}). \qquad (6.11)$$

Разложим функцию в ряд Тейлора в окрестности точки x_{n+1} с точностью до члена второго порядка малости

$$x(t_n) = x(t_{n+1}) - hx'(t_{n+1}) + \frac{1}{2}h^2x''(\tau), \qquad (6.12)$$

где $t_n < \tau < t_{n+1}$.

Отсюда

$$x(t_{n+1}) = x(t_n) + hx'(t_n) - \frac{1}{2}h^2x''(\tau). \qquad (6.13)$$

Следовательно,

$$\varepsilon_м = -\frac{1}{2}h^2x''(\tau). \qquad (6.14)$$

Таким образом, локальная методическая ошибка явного и неявного методов Эйлера определяется второй производной, следовательно, явный и неявный методы Эйлера представляют собой методы первого порядка точности. Поскольку в основе этих методов лежит известное разложение в ряд Тейлора, то их можно рассматривать как подкласс методов Тейлора - методы Тейлора первого порядка.

Очевидно, что точность методов первого порядка невысока, так что при их применении для обеспечения малой локальной методической ошибки следует интегрировать с малой величиной шага. Очевидно также, что локальная методическая ошибка метода трапе-

ций на порядок меньше в силу того, что методические ошибки явного и неявного методов Эйлера имеют противоположные знаки. Очевидно также, что при малых значениях вторых производных и малой величине шага локальная методическая ошибка может оказаться существенно меньше заданной. Поэтому представляется целесообразным по возможности увеличивать величину шага в процессе интегрирования.

Для оценки локальной методической погрешности ЯМЭ и НЯМЭ, как было показано, требуется определить вторую производную.

6.3. Вычисление второй производной

Вторую производную можно вычислить путем определения первых производных в крайних точках шага интегрирования на основе теоремы о среднем.

Вычисление иллюстрируется рис.6.1.

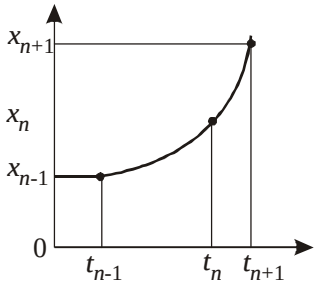


Рис.6.1. Иллюстрация вычисления второй производной

$$x'_n = \frac{x_n - x_{n-1}}{t_n - t_{n-1}}; \quad x'_{n+1} = \frac{x_{n+1} - x_n}{t_{n+1} - t_n};$$

$$x''_{n+1} = \frac{x'_{n+1} - x_n}{t_{n+1} - t_n}.$$

Если принять $t_{n+1} - t_n = t_n - t_{n-1} = h$, то выражение для второй производной будет иметь вид

$$x''_{n+1} = \frac{x_{n+1} - 2x_n + x_{n-1}}{h^2}. \qquad (6.15)$$

Подставив формулу (6.15) в формулу для локальной методической ошибки ЯМЭ и НЯМЭ:

$$\varepsilon_м = \pm\frac{1}{2}h^2 x''(\tau),$$

получим следующее выражение:

$$\varepsilon_{м} = \pm\frac{1}{2}\left(x_{n+1} - 2x_n + x_{n-1}\right). \qquad (6.16)$$

Представим один из возможных алгоритмов управления величиной шага интегрирования, основанный на требовании непревышения заданной локальной методической ошибки.

1. Пусть на n-м шаге интегрирования величина шага - h_n.
2. Определяем вектор локальной методической ошибки $\overline{E}_n$. Максимальная погрешность $\varepsilon_{\max} = \max\left|\overline{E}_{ni}\right|$.
3. Если $\varepsilon_{\max} > \varepsilon_{\text{заданное}}$, то шаг отбрасываем и пытаемся проинтегрировать, уменьшив шаг в два раза.

4. Если $\varepsilon_{\max} < \varepsilon_{\text{заданное}}$,

$$\begin{cases} \varepsilon_{\max} < 0{,}25\varepsilon_{\text{заданное}}, \text{ то } h_{n+1} = 2h_n; \\ 0{,}25\varepsilon_{\text{заданное}} \le \varepsilon_{\max} \le \varepsilon_{\text{заданное}}, \text{ то } h_{n+1} = h_n. \end{cases}$$

6.4. Анализ устойчивости методов численного интегрирования

6.4.1. Анализ устойчивости ЯМЭ

Можно показать, что глобальная ошибка для ЯМЭ в узле t_n умножается на величину $\|I - h*J\|$, называемую ***множителем перехода.*** Здесь I - единичная матрица, J - якобиан, //*// - некоторая матричная норма. Если не выполняется условие $\|I - h*J\| \le 1$, то метод Эйлера будет неустойчивым. Если в качестве нормы взять модуль множителя перехода, то условие будет выполнено, если для всех собственных значений λ матрицы J выполняется неравенство $|1 - h*\lambda| \le 1$. В случае одного вещественного уравнения это означает, что $h{\cdot}\lambda$ должно лежать в интервале (–2, 0). Отсюда следует, что при применении ЯМЭ на величину шага интегрирования накладывается ограничение: $h < 2/\lambda$.

6.4.2. Анализ устойчивости НЯМЭ

Можно показать, что для НЯМЭ множителем перехода является величина $\dfrac{1}{\|I + h*J\|}$. Анализ устойчивости выполняется аналогично. В случае одного вещественного уравнения

условием устойчивости является $\frac{1}{|1+h*\lambda|} \le 1$. Это условие выполняется всегда; следовательно, в отличие от явного метода Эйлера, ***неявный метод обладает глобальной устойчивостью.***

6.4.3. Анализ устойчивости метода трапеций

Метод трапеций будет устойчивым при условии, что его множитель перехода

$$\frac{\|I+0{,}5h*J\|}{\|I-0{,}5h*J\|} \le 1.$$

В случае одного вещественного уравнения условием устойчивости является

$$\frac{\|1+0{,}5h*\lambda\|}{\|1-0{,}5h*\lambda\|} \le 1.$$

Обратим внимание на то, что условия устойчивости должны выполняться для всех собственных значений Якобиана. Из этого следует, что при решении жестких уравнений (в случае, когда система дифференциальных уравнений описывает взаимодействующие процессы, одни из которых претерпевают быстрые изменения, а другие - медленные, например, апериодические), могут возникнуть существенные затруднения, связанные с выбором шага интегрирования. Для того чтобы отследить быстрые изменения, потребуется взять частую сетку, что ведет к большому объему вычислений. Однако когда вклад быстрых изменений в решение уменьшается, можно использовать более редкую сетку. Ряд численных методов нежелательно применять для решения жестких уравнений, так как при этом трудно гарантировать устойчивость. К таким методам, в частности, относится явный метод Эйлера. Он требует, чтобы максимальный шаг интегрирования не превышал удвоенной величины минимального собственного значения (постоянной времени) Якобиана. При этом общее время интегрирования должно задаваться с учетом максимального собственного значения (максимальной постоянной времени) того же Якобиана.

В заключение рассмотрим модели линейных реактивных элементов с учетом их конечно-разностного представления в случае применения неявного метода численного интегрирования Эйлера.

При использовании неявного метода интегрирования Эйлера компонентное уравнение для емкости $I_C = C\frac{dU_C}{dt}$ примет следующий вид:

$$i_C(t_{n+1}) = C\frac{U_C(t_{n+1}) - U_C(t_n)}{t_{n+1} - t_n} = \frac{C}{h}(U_C(t_{n+1}) - U_C(t_n)), \qquad (6.17)$$

где h - шаг интегрирования.

Конденсатор C можно представить также в виде дискретной схемной модели. Для этого преобразуем уравнение (6.17) к виду

$$i_{n+1} = \frac{C}{h}U_{n+1} - \frac{C}{h}U_n.$$

Здесь ток, протекающий через конденсатор, представлен сум-мой двух токов. Первая составляющая интерпретируется как резистор R с проводимостью $Y = \frac{C}{h}$, вторая - как источник тока $I = \frac{C}{h}U_n$. Следовательно, конденсатор C можно заменить дискретной моделью в виде параллельного соединения резистора и источника постоянного тока (рис.6.2).

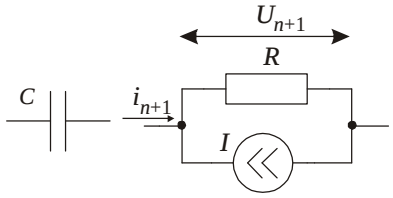


Рис.6.2. Дискретная схемная модель конденсатора

Отметим, что ток источника зависит от напряжения, имевшегося на предыдущем шаге интегрирования.

Аналогично компонентное уравнение для индуктивности $U_L = L\frac{dI_L}{dt}$ при использовании неявного метода Эйлера примет следующий вид:

$$U_L(t_{n+1}) = L\frac{i_L(t_{n+1}) - i_L(t_n)}{t_{n+1} - t_n} = \frac{L}{h}(i_L(t_{n+1}) - i_L(t_n)). \qquad (6.18)$$

Представим индуктивность в виде дискретной схемной модели. Для компактности уравнение (6.18) перепишем в виде

$$U_{n+1} = \frac{L}{h} i_{n+1} - \frac{L}{h} i_n \qquad (6.19)$$

и преобразуем его к виду

$$i_{n+1} = \frac{h}{L} U_{n+1} + \frac{h}{L} U_n . \qquad (6.20)$$

Первая составляющая уравне- ния (6.20) представляет собой ток, протекающий через резистор R с сопротивлением $R = \frac{L}{h}$, вторая - источник тока $I = \frac{h}{L} U_n$. Соответствующая схем-ная модель представлена на рис.6.3.

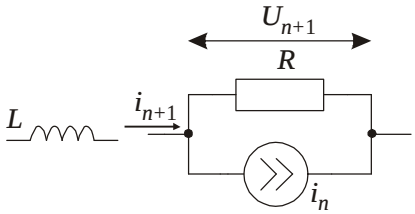


Рис.6.3. Дискретная схемная модель индуктивности

Лекция 7

Итерационные методы решения нелинейных уравнений

Наук зерно погребено
Под слоем пыли.
Кто не мудрит,
Тем путь открыт
Без их усилий.

И.В. Гёте. Фауст

Как было отмечено ранее (см. лекцию 4), при построении математической модели ИС чаще всего добиваются ее представления в виде системы обыкновенных дифференциальных уравнений (ОДУ). В процессе решения полученная система трансформируется, как правило, в систему нелинейных уравнений. Общий вид получаемой при этом системы уравнений

$$f(x)=0. \qquad (7.1)$$

В основе нахождения решений таких уравнений лежат две идеи:

1) линеаризации, т.е. замены исходной системы нелинейных уравнений приближенной - линейной;

2) последовательных приближений к точному решению - итераций: решение нелинейной задачи ищется как предел последовательности, члены которой получаются друг из друга по заданному алгоритму.

Для работы алгоритма требуется выбрать начальное приближение $x = x_0$ (желательно, достаточно близкое к решению). Что касается выбора начальных приближений, то для такой задачи не существует универсальных алгоритмов. Один из возможных способов выбора начальных приближений будет предложен ниже.

7.1. Идея итерации с неподвижной точкой

7.1.1. Алгоритм неподвижной точки

Большинство итерационных методов решения систем линейных и нелинейных уравнений могут быть рассмотрены как специальные случаи итерационного алгоритма с неподвижной точкой. Рассмотрим идею на примере уравнения с одним неизвестным.

Алгоритм неподвижной точки требует специальной формы записи

$$x = F(x). \qquad (7.2)$$

Целью алгоритма является нахождение такого $x = x^*$, которое сводит уравнение (7.2) к тождеству. Преобразуем уравнение (7.2) к виду

$$Y = x,\ Y = F(x). \quad (7.3)$$

Тогда геометрическая интерпре-тация алгоритма будет выглядеть следующим образом (рис.7.1).

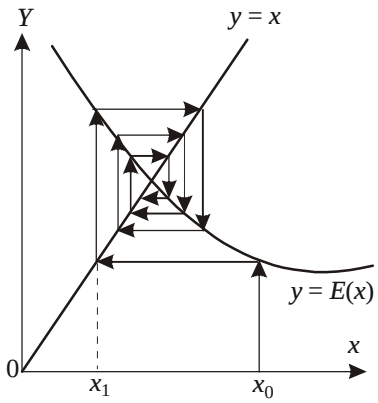


Рис.7.1. Геометрическая интерпретация алгоритма неподвижной точки

Предполагаем, что мы начинаем итерационную процедуру, выбрав $x = x_0$, в результате получаем x_1. Если $|x^* - x_1| < |x^* - x_0|$, то выбор начального приближения x_1 лучше, чем x_0. В качестве начального приближения выбираем полученное x_1 и повторяем итерации, пока не будет выполнено неравенство

$$|x_{k+1} - x_k < \varepsilon|, \qquad (7.4)$$

где ε - достаточно малая заданная величина.

В общем случае метод описывается рекурсивной формулой

$$x_{k+1} = F(x_k). \qquad (7.5)$$

Критерий, гарантирующий сходимость, определяется следующим образом (принцип сжатых отображений): если $f(x)$ есть сжатие n-мерного пространства R^n в R^n, т.е. существует константа $L<1$ (постоянная Липшица), такая что

$$\|f(y)-f(x)\| \le L\|y-x\|, \quad x,y \in R^n, \qquad (7.6)$$

то $f(y)$ имеет единственную неподвижную точку.

Последовательные итерации приводят к этой неподвижной точке. Если L близка к единице, то сходимость может быть очень медленной.

Методы неподвижной точки требуют, чтобы исходные уравнения $f(x) = 0$ записывались в стандартной форме (7.2)

$$F(x) = x - K(x)f(x). \qquad (7.7)$$

Здесь $K(x)$ - матричная неособенная функция от $f(x)$.

Ясно, что $K(x)$ может быть случайной функцией. Выбор вида функции $K(x)$ ведет к различным характеристикам сходимости. Отметим, что большинство итерационных методов решения систем нелинейных уравнений являются специальными случаями уравнения (7.7), причем каждый из методов характеризуется своим видом $K(x)$. Если, например, $K(x) = J^{-1}(x)$, где $J(x)$ - матрица Якоби

$$J = \begin{vmatrix} \frac{\partial f_1(x)}{\partial x_1} & \cdots & \frac{\partial f_1(x)}{\partial x_n} \\ \cdots & \cdots & \cdots \\ \frac{\partial f_n(x)}{\partial x_1} & \cdots & \frac{\partial f_n(x)}{\partial x_{n1}} \end{vmatrix}$$

то, подставляя это выражение в уравнение (7.7), получим:

$$x_{k+1} = x_k - J(x_k)^{-1}\Big|_{x=x_k} f(x_k). \qquad (7.8)$$

Это известная итерационная формула метода ***Ньютона***.

Если $K(x)$ принять за диагональную матрицу констант, то получим формулу метода простых итераций.

Метод Ньютона применяется на практике в большинстве случаев, поэтому рассмотрим его более подробно. Исходное решаемое уравнение $f(x) = 0$.

Известно, что всякую функцию $f(x)$ в окрестности решения можно разложить в ряд Тейлора. В этом случае

$$f(x) = f(x^*) + \left.\frac{\partial f(x)}{\partial x}\right|_{x=x^*} \cdot (x - x^*) + \\ + \frac{1}{2} \cdot \left.\frac{\partial^2 f(x)}{\partial x^2}\right|_{x=x^*} \cdot (x - x^*)^2 + \ldots = 0, \qquad (7.9)$$

где $x = x^*$ - решение.

Если в окрестности решения ограничиться двумя первыми членами разложения, то

$$f(x) = f(x^*) + \left.\frac{\partial f(x)}{\partial x}\right|_{x=x^*} \cdot (x - x^*) = 0 \quad . \qquad (7.10)$$

При этом локальная методическая погрешность

$$f(x) - f(x^*) = \frac{1}{2}\frac{\partial^2 f(x)}{\partial x^2} \cdot (x^k - x^*) \quad . \qquad (7.11)$$

Для метода Ньютона справедлива следующая теорема.

Теорема. Если $f(x^*) = 0$, производная $\left.\frac{\partial f(x)}{\partial x}\right|_{x=x^*} = 0$, а вторая производная $\frac{\partial^2 f(x)}{\partial x^2}$ непрерывна, то существует открытый интервал $N(x^*)$, содержащий x^* в решении, такой, что, если $x \in N(x^*)$, то для метода Ньютона x_k сходится к решению x^*, т.е. метод Ньютона гарантирует сходимость к решению при хорошем приближении. Следовательно, трудность применения метода Ньютона заключается в выборе начального приближения, которое находилось бы внутри интервала $N(x^*)$. Если x взят вне интервала окрестности решения (разложение в ряд Тейлора в окрестности решения), то нуль не будет найден. Возникает проблема сходимости (convergence problem).

Погрешность решения для k-й итерации

$$\varepsilon_k = x_k - x_k^* \quad . \qquad (7.12)$$

Разделив соотношение

$$f(x) = f(x^*) + \frac{\partial f(x)}{\partial x}\bigg|_{x=x^*} \cdot (x - x^*) + \frac{1}{2} \cdot \frac{\partial^2 f(x)}{\partial x^2}\bigg|_{x=x^*} \cdot (x - x^*)^2 = 0$$

на $\frac{\partial f(x)}{\partial x}$ и, воспользовавшись формулой Ньютона (7.8), получим:

$$\lim_{k\to\infty} \frac{\varepsilon_{k+1}}{\varepsilon_k^2} = \frac{1}{2} \frac{\frac{\partial^2 f(x)}{\partial x^2}}{\frac{\partial f(x)}{\partial x}}. \qquad (7.13)$$

Следовательно, если необходимо определить погрешность решения и сходимость, то нужно учитывать члены второго порядка малости в разложении функции в ряд Тейлора. При этом погрешность уменьшается от предыдущей итерации к последующей пропорционально квадрату. Таким образом, метод Ньютона имеет квадратичную сходимость.

Приведем пример квадратичной сходимости. Пусть на *k*-й итерации погрешность решения: $|\varepsilon_k| = \frac{1}{2}$.

Тогда

$$|\varepsilon_{k+1}| = \frac{1}{2^2}, \quad |\varepsilon_{k+2}| = \frac{1}{2^4}, \ldots, \quad |\varepsilon_{k+6}| = \frac{1}{2^{64}},$$

т.е. достаточно шести итераций для того, чтобы погрешность стала очень маленькой. На практике очень быстрая "ньютоновская" сходимость наблюдается нечасто, поскольку точным методом Ньютона удается воспользоваться сравнительно редко.

Примечание. Об итерационном процессе, для которого ошибка ε_k удовлетворяет соотношению

$$\lim_{k\to\infty} \frac{\varepsilon_{k+1}}{\varepsilon_k^p} = C \neq 0 \qquad (7.14)$$

говорят, что он имеет сходимость порядка *p*.

Поскольку метод Ньютона не гарантирует сходимость к решению, ему часто предшествует какой-либо глобально сходящийся алгоритм (например, метод деления отрезка пополам). В этом случае метод Ньютона является завершающей процедурой алгоритма, в котором на первых шагах применены более медленные, но надежные итерационные методы, позволяющие уменьшить интервал, в котором производится поиск решения, до значения радиуса сходимости (окрестности решения, гарантирующей сходимость метода Нью-

тона). Другой недостаток метода Ньютона заключается в том, что необходимо получать на каждой итерации новую матрицу Якоби. Формирование матрицы Якоби связано с решением задачи выбора алгоритма и составления программы вычисления частных производных, что является отдельной достаточно сложной задачей.

7.2. Метод Ньютона

Сократить время вычислений можно, избегая вычислений матрицы Якоби на каждой итерации. Если начальное приближение c на i-й итерации достаточно близко к решению, то можно на всех итерациях пользоваться матрицей Якоби, вычисленной для приближения для этой итерации. Эта модификация получила название ***огрубленного метода Ньютона***.

В практических алгоритмах, использующих метод Ньютона, применяются различные способы, улучшающие сходимость.

Для повышения надежности сходимости в методе Ньютона можно ввести преобразование, изменяющее невязку. Например, сохранять невязку, если она мала, и уменьшить ее, если она велика. Модификации метода (методы Ньютона - Рафсона) получают, изменяя приращение в формуле (7.8) путем умножения полученного приращения на некоторый коэффициент α, постоянный или меняющийся по некоторому алгоритму при переходе от итерации к итерации

$$x_{k+1} = x_k - \alpha J(x)^{-1}\Big|_{x=x_k} f(x_k). \quad (7.15)$$

В другом способе используется то обстоятельство, что вероятность сходимости увеличивается, если начальное приближение оказывается достаточно близко к решению. При этом само решение неизвестно, но можно предсказать характер его изменения при изменении некоторых параметров системы. В схемотехническом моделировании при анализе статического режима таким параметром является напряжение источника питания E. В этом случае алгоритм включает n шагов, за которые напряжение источника питания меняется от нуля до E, возрастая на каждом шаге на некоторое ΔE. Для первого шага задается нулевое начальное приближение и напряжение источника питания ΔE. Начальными приближениями для последующих шагов являются решения, полученные на предыдущих шагах.

В заключение еще раз отметим, что метод Ньютона в различных модификациях является основным методом, реализованным в современных системах схемотехнического моделирования.

В качестве примера рассмотрим применение метода Ньютона для анализа схемы делителя напряжения *V* из резистора *R* и диода *D* с математической моделью

$$I = I_0(e^{\frac{U}{m\varphi_0}} - 1).$$

Здесь I_0 - обратный тепловой ток диода; U - напряжение на диоде; φ_0 - температурный потенциал.

Схема делителя приведена на рис.7.2.

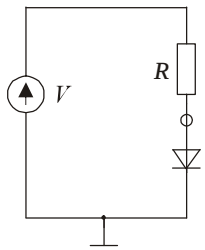


Рис.7.2. Схема делителя напряжения

Для данной схемы ММ, полученная методом узловых потенциалов, имеет вид

$$F(U) = \frac{V-U}{R} - I_0(e^{\frac{U}{m\varphi_T}} - 1) = 0. \quad (7.16)$$

Ее решение методом Ньютона

$$U_{i+1} = U_i + a^* D, \quad (7.17)$$

где

$$D = -\frac{\frac{V-U_i}{R} - I_0(e^{\frac{U_i}{m\varphi_T}} - 1)}{-\frac{1}{R} - \frac{I_0 e^{\frac{U_i}{m\varphi_T}}}{m\varphi_T}}.$$

Для решения уравнения (7.17) методом Ньютона необходимо задать начальное приближение U_0, коэффициент a^* и выполнять итерационные процедуры по формуле (7.17) до тех пор, пока не будет выполнено неравенство $|D| < \varepsilon$, где ε - заданная точность.

Лекция 8

Решение систем линейных алгебраических уравнений

Все опыт, опыт! Опыт это вздор.
Значенья духа опыт не покроет.
Все, что узнать успели до сих пор,
Искать не стоило и знать не стоит.

И.В. Гёте. Фауст

В лекции 7 было показано, что задача анализа переходных процессов в электронных схемах может быть сведена, в конечном итоге, к решению систем линейных алгебраических уравнений вида

$$Ax = b, \quad (8.1)$$

где A - заданная квадратная матрица порядка n; x - вектор-столбец неизвестных с n-компонентами; b - заданный вектор-столбец с n-компонентами. Будем полагать, что A - ***невырожденная*** матрица, т.е. ***матрица, детерминант которой не равен нулю***. Заметим, что вырожденная матрица не имеет обратной матрицы.

Методы решения таких систем можно разделить на три класса.

1. Класс ***точных методов***. Точные методы рекомендуется применять при решении систем линейных алгебраических уравнений (СЛАУ) небольшой размерности. Это связано с большим числом арифметических операций, требуемых для вычислений. Так, если матрица A неразреженная и имеет порядок n, то число необходимых операций при решении СЛАУ будет пропорционально n^3.
2. Класс ***итерационных методов***. Применяется при решении СЛАУ средней размерности, поскольку требует, как правило, меньших вычислительных затрат по сравнению с точными методами.
3. Класс ***вероятностных методов***. Применяется при решении СЛАУ большой размерности.

Приведем определение понятия ***точный метод***.

Метод решения системы уравнений относится к классу "точных", если в предположении отсутствия ошибок округления он дает точное решение задачи после конечного числа арифметических и логических операций.

Известно множество способов решения систем линейных уравнений, относящихся к классу "точных", таких как метод Гаусса, *LU*-факторизация, метод Краута.

8.1. Точные методы решения моделей линейных схем с хранимыми матрицами

Рассматриваемые далее алгоритмы ориентированы на хранимые неразреженные матрицы.

Хранимая матрица - матрица, все элементы которой хранятся в оперативной памяти машины.

Обычно ненулевые элементы разреженной матрицы хранятся в какой-либо специальной структуре данных или регенерируются по мере необходимости. Хранимая матрица также может иметь много ненулевых элементов, т.е. быть разреженной.

Частным случаем разреженной матрицы является ***ленточная матрица.*** Все ненулевые элементы ленточной матрицы расположены вблизи главной диагонали. Число диагоналей, на которых размещаются ненулевые элементы, называют ***шириной ленты.***

Для математических моделей интегральных схем характерны структуры матриц, подобные описанным.

Следует отметить, что численные методы, применяемые для хранимых матриц, более универсальны, и их можно модифицировать для работы с разреженными матрицами. Некоторые методы, применяемые в работе с разреженными матрицами, отличаются от методов, приспособленных к работе с хранимыми матрицами.

8.1.1. Метод Крамера

Широко известный метод ***Крамера***, выражающий каждый компонент решения отношением двух определителей, в настоящее время не применяется из-за сложности вычисления определителей. Все же у формул Крамера есть, по крайней мере, одно привлекательное свойство: в них компоненты решения вычисляются независимо друг от друга. По этой причине они могут оказаться практичными при распараллеливании решения задачи.

8.1.2. Обращение матрицы

Обращение матрицы - другой подход, математически привлекательный, но уязвимый в вычислительном отношении, заключается в том, что решение системы (8.1) записывается в виде $x = A^{-1} \cdot b$, где A^{-1} - обратная матрица. Однако практически в любом конкретном

приложении нет необходимости вычислять матрицу A^{-1} в явном виде. В качестве примера рассмотрим систему из одного уравнения: $7x = 21$.

Наилучший способ решения этой системы - деление: $x = 21/7 = 3$. Использование "обратной матрицы" привело бы к вычислению $x = (7 – 1)(21) = = (0{,}142857)(21) = 2{,}99997$.

Обращение числа 7 приводит к менее точному результату и требует дополнительного арифметического действия. В этом заключается главная причина, по которой рекомендуется избегать обращения матриц. Сказанное тем более справедливо для систем уравнений большой размерности.

8.1.3. Метод Гаусса (метод последовательного исключения неизвестных)

Рассмотрим более подробно один из класса точных методов решения систем линейных уравнений, а именно метод Гаусса, иначе называемый методом последовательного исключения неизвестных. Алгоритм Гауссова исключения в настоящее время занимает центральное место в численных методах решения систем линейных уравнений.

Идея метода основана на исключении переменных из уравнений системы по одной до тех пор, пока не останется только одна переменная в левой части одного уравнения. Затем данное уравнение решается относительно этой единственной переменной, и полученное значение подставляется в предыдущее уравнение для получения остающихся переменных. Очевидно, что предложенный алгоритм работает, если диагональный элемент $n_{ii} \neq 0$. Для того чтобы проиллюстрировать этот процесс, рассмотрим случай, когда размерность системы уравнений $n = 3$.

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 = b_1; \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 = b_2; \\ a_{31}x_1 + a_{32}x_2 + a_{33}x_3 = b_3. \end{cases} \qquad (8.2)$$

Алгоритм исключения Гаусса состоит из двух основных шагов:

1) шаг 1 (шаг прямого исключения);

2) шаг 2 (шаг обратного исключения).

Шаг 1 выполняется в $n – 1$ ступеней.

Ступень 1. Исключив переменную x_1 из второго и третьего уравнений системы (8.2), получим:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 = b_1; \\ 0 + a'_{22}x_2 + a'_{23}x_3 = b'_2; \\ 0 + a'_{32}x_2 + a'_{33}x_3 = b'_3. \end{cases} \qquad (8.3)$$

Первое уравнение системы (8.3) берется из системы (8.2). Второе уравнение системы (8.3) получено умножением первого уравнения этой системы на коэффициент $-a_{21}/a_{11}$ и сложением результата со вторым уравнением системы (8.2). Третье уравнение - путем умножения первого уравнения данной системы на коэффициент $-a_{31}/a_{11}$ и сложения результата с третьим уравнением системы (8.2).

Продолжим рассмотрение алгоритма.

Ступень 2. Исключив переменную x_2 из третьего уравнения системы (8.3), получим:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 = b_1; \\ 0 + a'_{22}x_2 + a'_{23}x_3 = b'_2; \\ 0 + 0 + a''_{33}x_3 = b''_3. \end{cases} \qquad (8.4)$$

Третье уравнение системы (8.4) получено умножением второго уравнения системы (8.3) на коэффициент $-a_{32}/a_{22}$ и сложением с третьим уравнением системы (8.3).

Описанные этапы приводят к уравнению вида

$$Ux = Mb,$$

где U - верхняя треугольная матрица. Диагональные элементы матрицы называются ***ведущими***. K-й ведущий элемент является коэффициентом при k-й переменной в k-м уравнении на k-м шаге исключения. В общем случае прямой ход состоит из $n-1$ шагов. Можно показать, что при прямом ходе требуется выполнить примерно $n^3/3$ операций умножения. Обратная подстановка требует приблизительно $n^2/2$ операций умножения. Для реальных значений n прямой ход доминирует в вычислительных затратах. Интуитивно можно утверждать, что k-й элемент не должен быть слишком малым, иначе при делении будут получаться очень большие числа с большими абсолютными погрешностями. В результате решение может сильно исказиться.

8.2. Способы уменьшения абсолютных погрешностей

Для уменьшения абсолютных погрешностей можно применить следующие способы.

1. ***Масштабирование коэффициентов***. Подход заключается в "отбрасывании" порядков при коэффициентах уравнений. При этом электрические параметры задаются не в единицах системы измерений СИ, а пропорционально им. Например, потен-

циал 0,0001 В (вольт) можно задать числом 10, т.е. вести расчет в милливольтах (мВ).

2. ***Метод Гаусса с выбором ведущего элемента***. Отличие его от вышеописанного подхода состоит в том, что на *k*-м шаге в качестве ведущего элемента берется ***наибольший по абсолютной величине*** элемент в неприведенной части *k*-го столбца. Строка, содержащая этот элемент, переставляется с *k*-й строкой. Таким же образом переставляются элементы правой части. Дальнейшее деление на ***наибольшее по абсолютной величине*** число ведет к наименьшему возрастанию ошибки округления.

Гауссово исключение с выбором ведущих элементов гарантированно дает малые ***невязки***. Связь между величиной ***ошибки и невязки*** отчасти определяется ***числом обусловленности.***

Дополнительная информация об ошибках, невязках и числе обусловленности приведена в приложении 2.

Лекция 9

Итерационные методы решения СЛАУ и систем нелинейных уравнений на основе алгоритма неподвижной точки

Наука эта - лес дремучий.
Исход единственный и лучший:
Профессору смотрите в рот
И повторяйте, что он врет.

И.В. Гёте. Фауст

В лекции 8 рассматривались "точные" методы решения систем линейных уравнений. Отмечалось, что их целесообразно применять при решении систем уравнений сравнительно небольшой размерности, поскольку число математических операций, требуемых для вычислений, пропорционально по меньшей мере кубу размерности системы. Применение итерационных методов (см. формулу 7.7)) в ряде случаев позволяет понизить общее число математических операций по сравнению с точными методами. Кроме того, методы итераций могут оказаться предпочтительнее с точки зрения устойчивости вычислений, в смысле влияния вычислительных погрешностей на результаты расчетов (см. приложение 1).

Различные подходы к выбору матрицы $K(x)$ (см. формулу 7.7)) приводят к различным методам. Наиболее известны: метод Якоби, метод Гаусса - Зейделя, метод последовательной верхней релаксации. Все перечисленные методы относятся к классу релаксационных.

9.1. Метод Якоби.

Линейный и нелинейный случаи

Метод Якоби можно отнести к методам простой итерации.

В основе методов простой итерации для линейного случая лежит преобразование уравнения

$$Ax = b \qquad (9.1)$$

к виду

$$x = Px + g \qquad (9.2)$$

и использование итерационной процедуры

$$x^{k+1} = Px^k + g, \qquad (9.3)$$

где P может быть произвольной *матрицей*. Сходимость итерационного процесса гарантируется следующими теоремами.

Теорема о достаточном условии сходимости метода простой итерации. Если $\|P\| < 1$, то система уравнений (9.2) имеет единственное решение и итерационный процесс сходится к решению со скоростью геометрической прогрессии.

Теорема о достаточном и необходимом условии сходимости метода простой итерации. Итерационный процесс сходится к решению при любом начальном приближении тогда и только тогда, когда все собственные значения матрицы P по модулю меньше единицы.

Метод Якоби для случая системы линейных уравнений представим следующей системой:

$$\begin{aligned}
&a_{11}x_1^{k+1} + a_{12}x_2^k + a_{13}x_3^k + \ldots + a_{1n}x_n^k = b_1;\\
&a_{21}x_1^k + a_{22}x_2^{k+1} + a_{23}x_3^k + \ldots + a_{2n}x_n^k = b_2;\\
&a_{31}x_1^k + a_{32}x_2^k + a_{33}x_3^{k+1} + \ldots + a_{3n}x_n^k = b_3;\\
&\ldots \qquad\qquad \ldots \qquad\qquad \ldots\\
&a_{n1}x_1^k + a_{n2}x_2^k + a_{n3}x_3^k + \ldots + a_{nn}x_n^{k+1} = b_n.
\end{aligned} \quad (9.4)$$

Тогда

$$x_1^{k+1} = \frac{1}{a_{11}}\left(b_1 - \sum_{i-1}^{n} a_{1i}x_i^k\right); \; x_2^{k+1} = \frac{1}{a_{22}}\left(b_2 - \sum_{i-2}^{n} a_{2i}x_i^k\right) \text{ и т.д.} \qquad (9.5)$$

Для нелинейного случая справедливо

$$\begin{aligned}
&F_1(x_1^{k+1}, x_2^k, x_3^k, \ldots, x_n^k) = 0;\\
&F_2(x_1^k, x_2^{k+1}, x_3^k, \ldots, x_n^k) = 0;\\
&F_3(x_1^k, x_2^k, x_3^{k+1}, \ldots, x_n^k) = 0;\\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\\
&F_n(x_1^k, x_2^k, x_3^k, \ldots, x_n^{k+1}) = 0.
\end{aligned} \qquad (9.6)$$

Нетрудно видеть, что каждое из уравнений систем (9.4) и (9.6) включает только одну неизвестную переменную. Можно считать, что метод Якоби преобразует исходную систему линейных или нелинейных уравнений относительно n неизвестных в n уравнений с одним неизвестным в каждом уравнении. Полученные уравнения решают одним из известных методов решения нелинейных уравнений, например методом Ньютона. При этом необходимо вычислять только n частных производных для n одномерных уравнений вместо $n \times n$ частных производных, если бы метод Ньютона был применен к исходной системе.

9.2. Метод Гаусса - Зейделя
(метод последовательных замещений)

9.2.1. Линейный и нелинейный случаи

Метод Гаусса - Зейделя основан на том, что *i*-е уравнение системы решается относительно *i*-й компоненты нового вектора *x*, причем для всех остальных компонентов вектора берутся значения, вычисленные ранее. Иначе говоря, в основе метода в линейном случае лежат уравнения вида

$$(L+D)x^{k+1} = -Ux^k + g, \quad (97)$$

где *L*, *U* - соответственно нижнее и верхнее треугольное разложение исходной матрицы *A* в формуле (9.1); *D* - диагональная матрица.

Без данного преобразования формулы (9.1) методом Гаусса - Зейделя итерационные уравнения системы выглядят следующим образом:

$$\begin{aligned}
&a_{11}x_1^{k+1} + a_{12}x_2^k + a_{13}x_3^k + \ldots + a_{1n}x_n^k = b_1;\\
&a_{21}x_1^{k+1} + a_{22}x_2^{k+1} + a_{23}x_3^k + \ldots + a_{2n}x_n^k = b_2;\\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\\
&a_{n1}x_1^{k+1} + a_{n2}x_2^{k+1} + a_{n3}x_3^{k+1} + \ldots + a_{nn}x_n^{k+1} = b_n.
\end{aligned} \quad (9.8)$$

Для случая системы нелинейных уравнений имеем

$$\begin{aligned}
&F_1(x_1^{k+1}, x_2^k, x_3^k, \ldots, x_n^k) = 0;\\
&F_2(x_1^{k+1}, x_2^{k+1}, x_3^k, \ldots, x_n^k) = 0;\\
&F_3(x_1^{k+1}, x_2^{k+1}, x_3^{k+1}, \ldots, x_n^k) = 0;\\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\\
&F_n(x_1^{k+1}, x_2^{k+1}, x_3^{k+1}, \ldots, x_n^{k+1}) = 0.
\end{aligned} \quad (9.9)$$

Недостатком метода Якоби и метода Гаусса - Зейделя является низкая скорость сходимости. Ускорения сходимости процесса Гаусса - Зейделя можно добиться путем применения метода поверхностной верхней релаксации.

9.3. Метод поверхностной верхней релаксации

Идея метода состоит в том, что приращение, полученное в результате одной итерации по методу Гаусса - Зейделя, умножается на некоторый релаксационный множитель и прибавляется к текущему значению.

$$x^{k+1} = x^k + \Omega \left| x_{\text{ГЗ}}^{k+1} - x^k \right|. \quad (9.10)$$

Здесь Ω - диагональная матрица параметров релаксации; $x_{\text{ГЗ}}^{k+1}$ - приращение, полученное по методу Гаусса - Зейделя.

$$\Omega = \begin{vmatrix} \omega_1 & 0 & 0 & \dots \\ 0 & \omega_2 & 0 & \dots \\ 0 & 0 & \dots & \dots \\ \dots & \dots & \dots & \omega_n \end{vmatrix}.$$

Можно доказать, что ***общим условием сходимости*** трех рассмотренных выше методов является требование того, чтобы все собственные значения соответствующих матриц были по модулю < 1.

Анализируя эти условия, интуитивно можно предположить, что для сходимости метода Якоби матрица A уравнения (9.1), должна быть близка к диагональной, а для сходимости метода Гаусса - Зейделя - почти нижней треугольной формы, т.е. условием сходимости обоих методов является преобладание диагональных элементов.

Ниже приведена графическая иллюстрация метода Гаусса - Зейделя для решения системы двух линейных уравнений. На рис.9.1 показаны случаи,

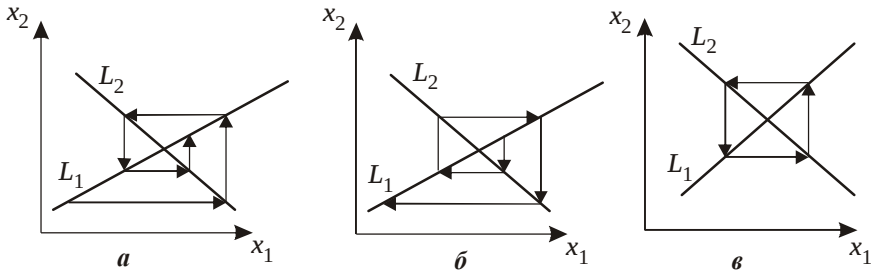


Рис.9.1. Иллюстрация метода Гаусса - Зейделя:
а - метод сходится; *б* - расходится; *в* - цикл

когда метод Гаусса сходится (рис.9.1,*а*), расходится (рис.9.1,*б*) и имеет цикл (рис.9.1,*в*). Здесь L_1, L_2 - графики зависимости x_2 от x_1 для первого и второго уравнений системы соответственно.

Сравнивая рис.9.1,*а* и 9.1,*б* можно видеть, что сходимость метода Гаусса - Зейделя может изменять характер при перестановке уравнений.

Для метода Гаусса - Зейделя выполняется следующая теорема о сходимости.

Теорема. Пусть A - вещественная симметричная положительно определенная матрица. Тогда метод Гаусса - Зейделя сходится. Оценим скорость сходимости метода Гаусса - Зейделя для схемы (рис.9.2).

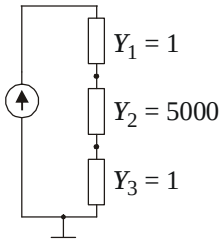


Рис.9.2. Принципиальная электрическая схема делителя напряжения

Общий вид математической модели схемы

$$\begin{cases} a_{12}x_1 + a_{12}x_2 = b_1; \\ a_{21}x_1 + a_{22}x_2 = b_2. \end{cases} \qquad (9.11)$$

Итерационная процедура метода Гаусса - Зейделя для данной математической модели

$$\begin{cases} a_{12}x_1^{k+1} + a_{12}x_2^{k} = b_1; \\ a_{21}x_1^{k+1} + a_{22}x_2^{k+1} = b_2. \end{cases} \qquad (9.12)$$

Точные значения x_1 и x_2:

$$x_1 = \frac{b_1 - a_{12}x_2}{a_{11}};\ x_2 = \frac{b_2 - a_{21}x_1}{a_{22}}. \qquad (9.13)$$

Из системы уравнений (9.12) получим следующие уравнения:

$$\begin{aligned} x_1^{k+1} &= \frac{b_1 - a_{12}x_2^{k}}{a_{11}}; \\ x_2^{k+1} &= \frac{b_2 - a_{21}x_1^{k+1}}{a_{22}}. \end{aligned} \qquad (9.14)$$

Погрешность решения для $k + 1$ итерации:

$$\begin{aligned} \Delta x_1^{k+1} &= x_1 - x_1^{k+1}; \\ \Delta x_2^{k+1} &= x_2 - x_2^{k+1}. \end{aligned} \qquad (9.15)$$

Подставим в формулы погрешности выражения уравнений (9.13) и (9.14)

$$\Delta x_1^{k+1} = \frac{1}{a_{11}}(b_1 - a_{12}x_2) - \frac{1}{a_{11}}(b_1 - a_{12}x_2^k) =$$
$$= \frac{1}{a_{11}}\left[a_{12}(x_2 - x_2^k)\right] = -\frac{a_{12}}{a_{11}}(x_2 - x_2^k) = -\frac{a_{12}}{a_{11}}\Delta x_2^k.$$

Аналогично для Δx_2^{k+1}

$$\Delta x_2^{k+1} = \frac{1}{a_{22}}(b_2 - a_{21}x_1) - \frac{1}{a_{22}}(b_2 - a_{21}x_1^{k+1}) =$$
$$= \frac{a_{21}}{a_{22}}(x_1 - x_1^{k+1}) = -\frac{a_{21}}{a_{22}}\Delta x_1^{k+1}.$$

Таким образом, выражения для погрешностей примут вид

$$\Delta x_1^{k+1} = -\frac{a_{12}}{a_{11}}\Delta x_2^k;\ \Delta x_2^{k+1} = -\frac{a_{21}}{a_{22}}\Delta x_1^{k+1}. \qquad (9.16)$$

Из выражения (9.16) получим

$$\Delta x_1^{k+1} = -\frac{a_{12}}{a_{11}}(\frac{a_{21}}{a_{22}})\Delta x_1^k = \frac{a_{12}}{a_{11}}\cdot\frac{a_{21}}{a_{22}}\cdot\Delta x_1^k;$$
$$\Delta x_2^{k+1} = \frac{a_{21}}{a_{22}}(\frac{a_{12}}{a_{22}})\Delta x_2^k.$$

Отсюда следует, что

$$x_1^{k+1} = \left(\frac{a_{12}}{a_{11}}\cdot\frac{a_{21}}{a_{22}}\right)^{k+1}\cdot\Delta x_1^0;\quad x_2^{k+1} = \left(\frac{a_{21}}{a_{22}}\cdot\frac{a_{12}}{a_{22}}\right)^{k+1}\cdot\Delta x_2^0. \qquad (9.17)$$

Условие сходимости итерационного процесса к решению

$$\left|\frac{a_{12}}{a_{11}}\cdot\frac{a_{21}}{a_{22}}\right| < 1.$$

Нетрудно видеть, что сходимость уравнений гарантируется при

$$\begin{cases} |a_{12}| < |a_{11}| \\ |a_{21}| < |a_{22}| \end{cases} \text{ либо } \begin{cases} |a_{21}| < |a_{22}| \\ |a_{12}| < |a_{22}| \end{cases}.$$

Для приведенного примера

$$\varphi_1(Y_1 + Y_2) - \varphi_2 Y_2 - EY_1 = 0;$$
$$\varphi_2(Y_2 + Y_3) - \varphi_1 Y_2 = 0;$$
$$a_{11} = Y_1 + Y_2; a_{12} = -Y_2;$$
$$a_{21} = -Y_2; a_{22} = Y_2 + Y_3;$$

$$\left(\frac{-Y_2}{Y_1 + Y_2} \cdot \frac{-Y_2}{Y_2 + Y_3}\right)^k = \left(\frac{1}{\frac{Y_1 + Y_2}{Y_2}} \cdot \frac{1}{\frac{Y_2 + Y_3}{Y_2}}\right)^k =$$

$$= \left(\frac{1}{\frac{Y_1}{Y_2} + 1} \cdot \frac{1}{\frac{Y_3}{Y_2} + 1}\right)^k = \left(\frac{1}{\frac{Y_1}{Y_2} + \frac{Y_3}{Y_2}} \cdot \frac{1}{\frac{Y_1 + Y_3}{Y_2} + 1}\right)^k = 0{,}99998^k.$$

При заданных значениях Y_1 = 1, Y_3 = 1, Y_2 = 5000 скорость сходимости к решению $0{,}99998^k$. Отметим, что при получении ММС для данного примера следует применять МУП. При этом формируется структурно-симметричная матрица узловых потенциалов. Следуя вышеприведенной теореме о сходимости, делаем вывод: при произвольном выборе номеров узлов принципиальной электрической схемы в получаемой системе уравнений характер сходимости метода Гаусса - Зейделя не меняется, т.е. метод Гаусса - Зейделя гарантирует сходимость.

Лекция 10

Анализ многошаговой формулы интегрирования. Метод простых итераций. Метод ускоренных итераций. Итерации Ньютона - Рафсона. Обратные итерации

Теория, мой друг, суха,
Но зеленеет жизни древо.
И.В. Гёте. Фауст

В лекции 4 было отмечено, что модифицированный метод узловых потенциалов заключается в алгебраизации исходных систем обыкновенных дифференциальных уравнений и их последующей линеаризации. Некоторые подходы к алгебраизации были рассмотрены в лекциях 5, 6, а линеаризации - в лекциях 7 - 9. В лекции 5 было отмечено, что для перехода от t_n к t_{n+1} желательно использовать значения решений (а возможно, и производных), вычисленные в нескольких предыдущих моментах времени (многошаговые методы). По сравнению с одношаговыми методами многошаговые могут быть более эффективны при той же точности, либо более точными при том же объеме вычислений.

Общий вид линейного многошагового метода

$$x_{n+1} - h\beta_0 f(x_{n+1}, t_{n+1}) - \sum_{i=1}^{k} (\alpha_i x_{n+1-j} + h\beta_1 f_{n+1-i}) = 0 \quad . \qquad (10.1)$$

Константы α_i и β_i подбираются априори. Если $\beta_0 = 0$, то метод будет явным; ненулевое β_0 приводит к неявному методу. В явных многошаговых методах для получения приближенного решения в момент времени t_{n+1}, вне зависимости от числа шагов в формуле, требуется знание только одного значения функции в предыдущий момент времени. При использовании неявных методов интегрирования ОДУ возникают нелинейные алгебраические уравнения. При этом требуется применение итерационных процедур на каждом шаге интегрирования.

В большинстве неявных многошаговых методов реализованы алгоритмы решения систем нелинейных уравнений, рассмотренных ранее. В настоящее время многошаговые методы являются стандартным средством решения задач динамического анализа интегральных схем. Однако при их реализации приходится сталкиваться с рядом трудностей.

1. При использовании трехшагового метода требуется знание решения, полученного в двух предыдущих моментах времени. При этом на первом шаге интегрирования известны

лишь начальные условия. Один из вариантов, позволяющих обойти эти затруднения, - воспользоваться одношаговым методом, чтобы получить требуемые предыдущие решения, и только после этого применять трехшаговый метод.

2. Многошаговые методы в соответствии с формулой (10.1) могут применяться только с постоянным шагом h, в то время как одношаговый метод допускает использование переменного шага.

Достоинства, присущие многошаговым методам, позволили создать программы, преодолевающие отмеченные трудности. В некоторых из них используются чисто эвристические приемы. Раздел, посвященный этим методам, имеется почти в каждом учебнике по численным методам.

10.1. Устойчивость многошаговых методов

Многошаговые методы хороши тем, что среди них можно найти методы любого высокого порядка. Вместе с тем, реализовав явный многошаговый метод, можно убедиться в его неустойчивости для решения многих устойчивых задач при произвольном выборе шага интегрирования. К сожалению, подобное поведение наблюдается и у неявных методов.

Установлено, что никакой неявный многошаговый метод не может быть абсолютно устойчив, если его порядок выше второго. Определено также, что при решении жестких систем уравнений целесообразно применять методы Гира.

10.2. Сходимость линейных многошаговых методов

Обратимся к общему виду линейного многошагового метода. Проанализируем сходимость решений нелинейных алгебраических уравнений, получаемых на каждом шаге численного интегрирования при использовании некоторых итерационных методов.

Итак, требуется решить неявное уравнение вида (10.1).

Так как член под знаком суммы известен, то обозначим его как ω_n; тогда формула (10.1) примет вид

$$x_{n+1} - h\beta_0 f(x_{n+1}, t_{n+1}) - \omega_n = 0. \quad (10.2)$$

В общем случае формула (10.2) является системой нелинейных трансцендентных уравнений. Задача заключается в определении x_{n+1}. Рассмотрим некоторые варианты решения этой задачи.

1. ***Метод простых итераций (метод Якоби)***. Формула метода простых итераций (см. лекцию 9)

$$x_{n+1}^{(s+1)} - h\beta_0 f(x_{n+1}^{(s)}, t_{n+1}) - \omega_n = 0. \quad (10.3)$$

Пусть x^* - точное решение уравнения (10.2), тогда

$$x_{n+1}^{*} - h\beta_0 f(x_{n+1}^{*}, t_{n+1}) - \omega_n = 0. \quad (10.4)$$

Вычитая из уравнения (10.3) уравнение (10.4), получим:

$$x_{n+1}^{(s+1)} - x_{n+1}^{*} = h\beta_0 (f_{n+1}^{(s)} - f_{n+1}^{*}). \quad (10.5)$$

Применяя теорему о среднем, преобразуем формулу (10.5):

$$x_{n+1}^{(s+1)} - x_{n+1}^{*} = h\beta_0 \left[f_x^{'}\right]_{\vartheta} (x_{n+1}^{(s)} - x_{n+1}^{*}), \quad (10.6)$$

где $x_{n+1}^{(s+1)} \le \vartheta \le x_{n+1}^{*})$.

По условию Липшица $f_x^{'} < L$, тогда

$$\left\| x_{n+1}^{(s+1)} - x_{n+1}^{*} \right\| \le h\beta_0 L \left\| x_{n+1}^{(s)} - x_{n+1}^{*} \right\|. \quad (10.7)$$

По индукции

$$\left\| x_{n+1}^{(s+1)} - x_{n+1}^{*} \right\| \le (h\beta_0 L)^{(s+1)} \left\| x_{n+1}^{[0]} - x_{n+1}^{*} \right\|. \quad (10.8)$$

Принимая во внимание теорему о единственности решения, необходимое и достаточное условие сходимости итерационного процесса метода Якоби имеет вид:

$$\left| h\beta_0 L \right| < 1. \quad (10.9)$$

Так как $L \le |\lambda_{\max}|$, где $\lambda_{\max}$ - наибольшее собственное значение матрицы f_x', итерационный процесс Якоби сходится к единственному решению при

$$h\beta_0|\lambda_{\max}|<1. \qquad (10.10)$$

Для быстрой сходимости необходимо потребовать, чтобы

$$h\beta_0|\lambda_{\max}|<<1. \qquad (10.11)$$

Следовательно, условия сходимости зависят от величины *h*. Если $|\lambda_{\max}|$ велико, то *h* должно быть очень мало.

2. ***Метод ускоренных итераций.*** Метод ускоренных итераций является модификацией метода итераций Якоби. Он описывается формулой:

$$(1+\alpha)x_{n+1}^{(s+1)} - h\beta_0 f(t_{n+1}, x_{n+1}^{(s)}) - \omega_n - \alpha x_{n+1}^{(s)} = 0. \qquad (10.12)$$

Здесь α - параметр ускорения.

Если $\alpha = 0$, то получаем метод простых итераций.

Условия сходимости определим аналогично тому, как их определяли для метода простых итераций.

Точное решение для метода ускоренных итераций принимает вид

$$(1+\alpha)x_{n+1}^{*} - h\beta_0 f(t_{n+1}, x_{n+1}^{*}) - \omega_n - \alpha x_{n+1}^{*} = 0. \qquad (10.13)$$

Вычитая формулу (10.13) из формулы (10.12) и пользуясь теоремой о среднем, получаем

$$x_{n+1}^{(s+1)} - x_{n+1}^{*} = \left(\frac{h\beta_0 \left[f_x' \right]_0 + \alpha I}{1+\alpha} \right) (x_{n+1}^{(s)} - x_{n+1}^{*}). \qquad (10.14)$$

Условие сходимости

$$\left\| \frac{h\beta_0 \left[f_x' \right]_0 + \alpha I}{1+\alpha} \right\| < 1 \quad \text{или} \quad \frac{h\beta_0 \lambda_{\max} + \alpha I}{1+\alpha} < 1. \qquad (10.15)$$

Здесь I - единичная матрица.

3. ***Итерационный метод Ньютона - Рафсона.*** Метод описывается формулой

$$x_{n+1}^{(s+1)} = x_{n+1}^{(s)} - \left[I - h\beta_0 J_{n+1}^{(s)}\right]^{-1}\left[h\beta_0 f_{n+1}^{(s)} - x_{n+1}^{(s)} - \omega_n\right]. \quad (10.16)$$

Здесь $J_{n+1}^{(s)}$ - матрица Якоби $f_x^{'}$, оцененная в точке $x(t_n)$, однократное применение итерации соответствует решению параметризованной формы. Найдем условие сходимости.

Следуя вышеприведенной последовательности действий, получим:

$$\left\| \left[I - h\beta_0 J_{n+1}^{(s)}\right]^{-1} \right\| \cdot \left\| \left(\frac{d}{dx}\right)\left[I - h\beta_0 J_{n+1}^{(s)}\right] \right\| \cdot \left\| x_{n+1}^{s+1} - x_{n+1}^{(s)} \right\| \le 1 . \quad (10.17)$$

Недостатком применения метода Ньютона является то, что на каждой итерации приходится пересчитывать значения элементов матрицы Якоби $J_{n+1}^{(s)}$.

4. ***Обратные итерации.*** Рассмотренные выше методы можно отнести к методам прямых итераций, так как они осуществлялись одинаково, а именно: задавалось начальное приближение, подставлялось в правую часть рекуррентного выражения, затем вычислялось новое приближение и подставлялось в правую часть и т.д.

Аналогично можно сформировать уравнения с обратными итерациями в виде

$$x_{n+1}^{(s)} = h\beta_0 f(t_{n+1}, x_{n+1}^{(s+1)}) + \omega_n , \quad (10.18)$$

которые требуют решения неявных уравнений.

Следуя обычной процедуре, запишем:

$$x_{n+1}^{(s)} - x_{n+1}^{*} = h\beta_0\left[f_{n+1}^{(s+1)} - f_{n+1}^{*}\right] = h\beta_0\left[f_x^{'}\right]_{\vartheta}(x_{n+1}^{(s+1)} - x_{n+1}^{*}) .$$

Условие сходимости:

$$\left(\frac{1}{h\beta_0}\right)\left\|\left[f_x^{'}\right]^{-1}\right\| < 1 \quad \text{или} \quad \left(\frac{1}{h\beta_0}\right)\lambda_{\max}\Big|^{-1} < 1 .$$

Этим выражением устанавливается нижняя граница для h.

Как следует из формул (10.3) и (10.12), метод итераций Якоби и метод ускоренных итераций легко реализуются, но сходимость зависит от максимального собственного значения матрицы Якоби λ_{max} (см. формулы (10.10), (10.15)). Из этих формул следует, что если значение $|\lambda_{max}|$ | велико, то шаг следует выбирать малым.

Условиям сходимости метода Ньютона посвящено много литературы. Итерации Ньютона имеют меньшую область сходимости, чем простые и ускоренные. Зато обратные итерации имеют громадную область сходимости из-за наличия нелинейной границы на h, но существует проблема решения неявных уравнений.

Приведем рекомендации для выбора метода решения задачи.

Если число обусловленности *матрицы Якоби* меньше 10, рекомендуется применять простые или ускоренные итерации. В противном случае желательно использовать итерационный метод Ньютона или методы обратных итераций с выбором шага на основе желаемого числа итераций на шаге. Оптимальное число итераций в корректирующей формуле - 2 (две).

1. Если корректирующая формула в методе не итерируется, то устойчивость метода зависит как от предсказывающих, так и от корректирующих формул.

2. Если корректирующая формула итерируется, то нет уверенности, что устойчивость зависит от корректирующей формулы.

В заключение отметим, что современное математическое обеспечение САПР БИС реализует решение математических моделей, являющихся системами обыкновенных дифференциальных уравнений. На каждом шаге интегрирования требуется решать систему нелинейных алгебраических уравнений с помощью метода Ньютона или других итерационных методов.

Лекция 11

Алгоритмы решения математических моделей БИС по постоянному току

Комары и мошкара,
Захотели взбучки?
Вправду ли вы мастера
Или недоучки?
И.В. Гёте. Фауст

Анализ электронных схем по постоянному току выполняется для получения статических вольт-амперных характеристик электронных схем и их компонентов. Математические методы решения этой задачи частично были рассмотрены в лекциях, посвященных решениям систем обыкновенных дифференциальных уравнений, систем нелинейных трансцендентных уравнений, систем линейных уравнений.

В данной лекции автор предлагает три подхода к решению поставленной задачи.

Первый подход заключается в ***решении систем уравнений вида***

$$F(x)=0, \quad (11.1)$$

получаемых в случае, если анализируется схема, включающая нелинейные элементы при условии отсутствия элементов, параметры которых изменяются во времени.

Второй подход - это так называемый ***анализ статики через динамику***. Подход основан на представлении внешних источников ***постоянного напряжения или тока импульсами "бесконечно большой" длительности*** и последующем анализе переходных процессов в схеме при воздействии на нее данных импульсов.

Математическая модель в этом случае имеет вид

$$F(x,\frac{dx}{dt},t)=0 \quad (11.2)$$

с дополнительным условием: $\frac{dx}{dt}=0$ при $t\to\infty$.

Преимущество этого подхода заключается в том, что при получении решения на каждом шаге численного интегрирования неявным методом (например, неявным методом Эй-

лера) в процессе алгебраизации получаются уравнения вида (11.1), для которых, как правило, удается подобрать достаточно хорошие начальные приближения к решению.

Это обеспечивается тем, что моделирование проводится в нескольких временных точках фронтов внешних импульсов, и в качестве начального приближения для последующей точки берутся решения, полученные для предыдущей. Кроме того, в случае анализа многостабильных схем (схем, которые могут находиться в нескольких устойчивых состояниях (например, триггеры), зависящих от формы и последовательности подаваемых сигналов, повышается достоверность моделирования.

Третий подход - ***решение экстремальной задачи.*** Он заключается в построении некоторой целевой функции Ф(x), где x^* - ***искомое решение статической*** задачи. Нахождение экстремума такой функции дает требуемое x^*.

Проиллюстрируем подход на примере определения потенциала узла 1 для схемы делителя, представленной на рис.11.1.

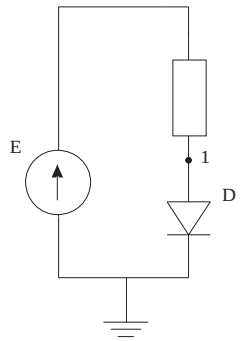


Рис.11.1. Принципиальная электрическая схема

Математическая модель схемы, составленная по МУП для узла 1, имеет вид

$$I_1(\varphi) - I_2(\varphi) = 0. \quad (11.3)$$

Если в качестве целевой функции взять модуль выражения (11.3), то можно видеть, что в точке решения будет экстремум (минимум) целевой функции $\Phi(x) = |I_1(\varphi) - I_2(\varphi)| = 0$. Графически это можно представить в следующем виде (рис.11.2).

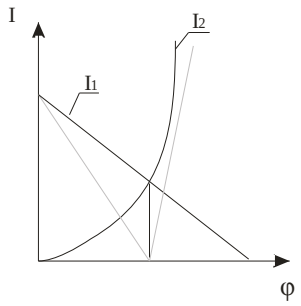


Рис.11.2. Иллюстрация третьего подхода

Известен ряд методов оптимизации, например методы сопряженных градиентов или направлений, применимых к симметричным положительно определенным матрицам. Методы относятся к классу "точных". При этом решение получается за конечное число шагов, но из-за ошибок округления методы рассматриваются как итерационные. Примеры подобных методов рассмотрены в лекции 14.

Первый и второй подходы, как было показано ранее, обычно приводят к решению систем линейных алгебраических уравнений (СЛАУ). Третий подход также часто требует решения таких систем.

Лекция 12

Многовариантный анализ. Статистический анализ. Анализ чувствительности

"Все кончено". А было ли начало?
Могло ли быть? Лишь видимость мелькала.
Зато в понятье вечной пустоты
Двусмысленности нет и темноты.

И.В. Гёте. Фауст

Все параметры P, характеризующие работу электронной схемы на этапе схемотехнического проектирования, можно разделить на входные (внешние), внутренние и выходные. Примером входного параметра может служить температура окружающей среды; внутреннего - сопротивление входящего в схему резистора, примером выходного - задержка прохождения сигнала в анализируемой схеме.

12.1. Параметрическая оптимизация

При разработке любой электронной схемы важно знать, как влияют изменения внутренних параметров компонентов схемы, в том числе в силу их технологического разброса, а также изменения состояния окружающей среды на выходные характеристики этой схемы.

Это позволяет, ***во-первых***, решить задачу детерминированной ***параметрической оптимизации*** электронных схем, которая обычно сводится к задаче поиска экстремума критерия оптимизации, называемого целевой функцией. Задача решается путем многократного выполнения требуемого вида анализа (многовариантный анализ). ***Целевая функция*** определяется как некоторая обобщенная функция выходных параметров (электрических характеристик), зависящая от внутренних и внешних параметров. Представим целевую функцию в виде

$$\Phi = \Phi[\varphi(P)]. \qquad (12.1)$$

Здесь Ф - критерий оптимизации; $(\varphi_1, \varphi_2, \ldots, \varphi_m)^T$ - вектор-столбец выходных параметров схемы (T - знак транспонирования); $P = (P_1, P_2, \ldots, P_n)$ - вектор-строка входных и

внутренних параметров схемы; m - число выходных параметров; n - число входных и внутренних параметров.

Примером постановки задачи оптимизации может служить поиск значения сопротивления R_2, обеспечивающего максимальную мощность, выделяемую на нем при заданном значении сопротивления R_2 (рис.12.1).

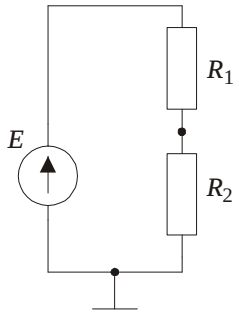


Рис.12.1. Принципиальная электрическая схема делителя напряжения

Для данного примера целевая функция имеет вид

$$\Phi_{\max}(R) = \left[\left(\frac{E}{R_1 + R_2} \right)^2 \cdot R_2 \right].$$

Большинство используемых методов поиска экстремума основано на вычислении на каждом шаге градиента многомерной функции Ф[φ(P)]. Обычно известна аналитическая зависимость критерия оптимизации от выходных параметров схемы. Поэтому градиент функции (12.1) может быть записан в виде

$$\frac{\partial \Phi}{\partial P} = \frac{\partial \Phi}{\partial \varphi} \frac{\partial \varphi}{\partial P}. \qquad (12.2)$$

Запись вида $\frac{\partial A}{\partial B}$, где A и B - векторы, содержащие N_A и N_B элементов соответственно, означает матрицу, состоящую из N_A строк и N_B столбцов.

Компоненты вектора $\frac{\partial \Phi}{\partial \varphi}$ вычисляются по аналитическим зависимостям как функции выходных параметров. Таким образом, расчет градиента функции Ф(P) сводится к опре-

делению матрицы коэффициентов $\frac{\partial \varphi}{\partial P}$ выходных параметров схемы по отношению к изменению входных и внутренних параметров.

Учет влияния изменения внутренних параметров компонентов схемы на ее выходные параметры позволяет сформулировать требования к допустимым отклонениям внешних и внутренних параметров от номинальных значений, при которых обеспечивается работоспособность схем, либо определить процент выхода годных изделий при заданном технологическом разбросе параметров. Эти задачи решаются в рамках статистического анализа.

12.2. Методы статистического анализа

В современных системах схемотехнического проектирования применяются следующие популярные методы статистического анализа.

1. **Метод наихудшего случая**. Этот метод постулирует (основан на предположении), что каждый из множества выбранных параметров, подверженных разбросу, отклоняется на максимально допустимую величину. Результирующее отклонение выходного параметра определяется как сумма ***модулей*** его отклонений по каждому из параметров множества.

Применение данного метода позволяет сформулировать требования к максимально допустимому разбросу внутренних параметров, который обеспечивает 100%-ный выход годных изделий.

Основной недостаток метода заключается в предположении реализации наихудшего случая сочетания изменения внутренних и внешних параметров. Слепо следуя этому методу, разработчик-схемотехник предъявит излишне жесткие требования к технологическим разбросам параметров компонентов разрабатываемой интегральной схемы. Сказанное подтверждает вышеприведенный пример определения допустимого разброса параметров для делителя напряжения на резисторах.

Реальные технологические отклонения внутренних параметров от требуемых (номинальных) значений подчинены известным законам распределения. Существование таких законов учитывает метод Монте-Карло.

2. **Метод Монте-Карло.** В методе Монте-Карло, в отличие от метода наихудшего случая, применяются законы распределения технологических отклонений значений внутренних параметров относительно их номинальных значений. Это может быть нормальный закон распределения, либо распределение Гаусса.

Для решения задач параметрической оптимизации и статистического анализа часто пользуются функциями чувствительности.

12.2.1. Анализ чувствительности

Определение 1. Функцией относительной (нормализованной) чувствительности, или чувствительностью параметра φ схемы относительно параметра *p*, обозначаемой *S*, называется функция, определенная следующим выражением:

$$S = \frac{\partial \varphi}{\varphi} \Big/ \frac{\partial p}{p}. \qquad (12.3)$$

Определение 2. Функцией абсолютной (ненормализованной) чувствительности, или чувствительностью параметра φ схемы относительно параметра *p*, называется функция, которая определяется как частная производная

$$S = \frac{\partial \varphi}{\partial p}. \qquad (12.4)$$

В случае, когда φ и *p* являются векторами, *S* становится матрицей. При этом элементы матрицы называются *коэффициентами чувствительности*.

12.2.2. Анализ чувствительности для составления критерия параметрической оптимизации и статистического анализа

Поскольку число параметров, по которым возможно оптимизировать схему или выполнять статистический анализ чрезвычайно велико, возникает необходимость ограничить это число до приемлемой величины. Знание коэффициентов чувствительности позволяет выделить то небольшое множество параметров, влияние которого будет учитываться при решении указанных выше задач. Наиболее явным является применение этих данных для оценки допусков на параметры компонентов схемы.

Рассмотрим ***простейший пример*** определения допустимого разброса параметров для делителя напряжения на резисторах (см. рис.12.1).

Пусть E = 1 В, R_1 = 2 Ом, R_2 = 1 Ом.

Потребуем, чтобы значение потенциала φ на нагрузке поддерживалось с точностью ±2%. В этом случае допуски на значения R_1 и R_2 можно приближенно оценить следующим образом.

При заданных значениях R_1, R_2

$$\varphi = E \cdot R_2/(R_1 + R_2) = 1/3 \text{ В}.$$

Отношение напряжений на делителе определяется выражением

$$T = R_2/(R_1 + R_2) = 1/3 \text{ В}.$$

Из формулы (12.1) получим нормализованную чувствительность относительно R_1

$$S_1 = -R_1/(R_1 + R_2) = -2/3;$$

нормализованную чувствительность относительно R_2

$$S_2 = R_1/(R_1 + R_2) = 2/3.$$

Если φ должно поддерживаться с точностью ±2%, то допуски на значения R_1 и R_2 можно приближенно оценить, исходя из следующего соотношения:

$$0{,}02 \geq \frac{\Delta T}{T} = \frac{\Delta\varphi}{\varphi} = S_1\frac{\Delta R_1}{R_1} + S_2\frac{\Delta R_2}{R_2} = \frac{2}{3}\left(-\frac{\Delta R_1}{R_1} + \frac{\Delta R_2}{R_2}\right).$$

Последнее соотношение будет выполнено со 100%-ной гарантией, если

$$\begin{cases} \dfrac{\Delta R_1}{R_1} \leq 0{,}015; \\ \dfrac{\Delta R_2}{R_2} \leq 0{,}015. \end{cases}$$

В этом случае в схеме делителя напряжения необходимо использовать резисторы с 1%-ной точностью.

Лекция 13

Расчет коэффициентов чувствительности

Видя, как еще я мал,
Он мне пыль в глаза пускал.
Как глубок его подлог,
Я тогда понять не мог.
В нынешнее время - дудки!
Не пройдут такие шутки.

И.В. Гёте. Фауст

Как было отмечено в лекции 12, многие задачи проектирования электронных схем связаны с определением степени влияния внешних и внутренних параметров на входные с последующим использованием полученных результатов. Для количественной оценки изменения выходных параметров при вариациях внутренних и внешних были введены понятия коэффициентов (функций) чувствительности (формулы (12.3), (12.4)).

Коэффициенты чувствительности можно определять несколькими способами: методом приращений, методом присоединенной цепи и методом определения производной.

13.1. Метод составления схемы в приращениях

В этом методе коэффициент чувствительности *j*-го выходного параметра к изменению *i*-го входного параметра определяется по формуле

$$\frac{\partial \varphi_j}{\partial P_i} \cong \frac{\Delta \varphi_j}{\Delta P_i} = \\ = \frac{\varphi_j(P_1, P_2, ..., P_{i-1}, P_i + \Delta P_1, P_{i+1}, ..., P_m,) - \varphi_j(P_1, P_2, ..., P_m)}{\Delta P_i}. \quad (13.1)$$

На практике каждая ветвь исходной электрической схемы заменяется эквивалентной схемой, в которой в качестве параметра фигурирует приращение изменяемого параметра ветви, а токи и напряжения ветвей заменяются приращениями токов и напряжений.

13.1.1. Пример построения эквивалентной схемы в приращениях для проводимости

Пусть дана электрическая схема проводимости Y (рис.13.1).

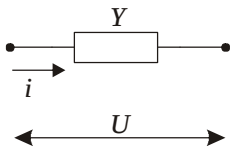


Рис.13.1. Электрическая схема проводимости

В соответствии с законом Ома ток через проводимость $i = Y \cdot U$.

Пусть параметр (проводимость) Y получает приращение ΔY (рис.13.2).

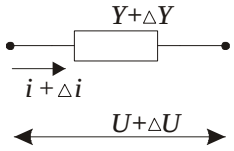


Рис.13.2. Электрическая схема измененной проводимости

В этом случае меняются ток проводимости и напряжение на ней. В соответствии с законом Ома

$$i + \Delta i = (Y + \Delta Y)(U + \Delta U) = Y \cdot U + Y \cdot \Delta U + \Delta Y \cdot U + \Delta Y \cdot \Delta U .$$

Считаем, что $\Delta Y \cdot \Delta U = 0$, т.е. имеет второй порядок малости.

Следовательно,

$$\Delta i = Y \cdot \Delta U + \Delta Y \cdot U . \qquad (13.2)$$

Эквивалентная схема проводимости в приращениях, т.е. схема, в которой токи и напряжения заменены их приращениями, будет выглядеть следующим образом (рис.13.3).

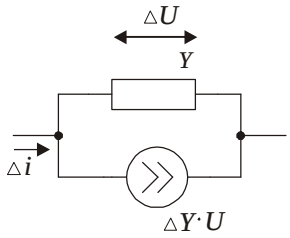


Рис.13.3. Эквивалентная схема проводимости в приращениях

Аналогично можно построить эквивалентные схемы в приращениях для всех базовых элементов библиотек схемотехнических моделей.

Достоинство метода приращений заключается в простоте реализации и в возможности определения коэффициентов чувствительности любых выходных параметров схемы.

Однако метод имеет и ряд существенных недостатков.

Во-первых, коэффициенты чувствительности определяются со значительной ошибкой, так как функции $\varphi(P)$, как правило, нелинейные. *Во-вторых*, для вычисления коэффициента чувствительности выходного параметра по m входным и внутренним параметрам требуется $(m + 1)$ раз рассчитывать схему. Это связано со значительными затратами машинного времени. Затраты времени особенно велики при расчете коэффициентов чувствительности динамических параметров схемы, для определения которых необходимо численное интегрирование системы дифференциальных уравнений.

Заметим, что большие затраты машинного времени для расчета градиента функции оптимизации препятствуют оптимизации сложных электронных схем при большой размерности пространства входных и внутренних параметров.

13.2. Метод присоединенной цепи

Метод присоединенной цепи позволяет анализировать как линейные, так и нелинейные схемы, содержащие компоненты всех известных типов. При этом для определения чувствительности нужно использовать результаты расчета двух схем - исходной и специальной вспомогательной - присоединенной. Присоединенная схема строится по формальным правилам путем некоторых преобразований элементов исходной схемы, т.е. основным требованием для присоединенной схемы является идентичность ее топологии с исходной.

В общем случае для получения присоединенной схемы необходимо закоротить все источники напряжения, на выходе схемы подключить единичный источник тока, использовать специальные модели, заменяющие нелинейные элементы. При этом чувствитель-

ность выходного напряжения вычисляется на основе расчета статического режима исходной и присоединенной схем. Анализ динамических коэффициентов чувствительности основан на анализе статических коэффициентов чувствительности для каждого шага интегрирования.

В основе подхода лежит ***теорема Теллиджена (Телегена).***

Теорема утверждает, что если $V_1(t), V_2(t), ..., V_{K_B}(t)$ - напряжение K_B ветвей, а $I_1(t), I_2(t), ..., I_{K_B}(t)$ - токи ветвей K_B некоторой цепи A, состоящей из произвольных сосредоточенных двухполюсников, то

$$\sum_{i=1}^{K_B} V_i(t) I_i(t) = 0 \qquad (13.3)$$

в любой момент времени.

Физический смысл выражения состоит в том, что сумма мгновенных энергий, передаваемых во все ветви, равна нулю. Эта формула выражает закон сохранения энергии в любой момент времени применительно к электрическим цепям.

Если имеются две схемы A и A' с одинаковой топологией (одинаковыми матрицами инциденций) и в общем случае различными характеристиками ветвей, то следствием теоремы Теллиджена являются следующие утверждения:

$$\sum_{i=1}^{K_B} V_i(t) I'_i(t) = 0; \qquad (13.4)$$

$$\sum_{i=1}^{K_B} V'_i(t) I_i(t) = 0. \qquad (13.5)$$

В формулах (13.4), (13.5) $V_1(t), V_2(t), ..., V_{K_B}(t)$ - напряжения ветвей, $I_1(t)$, $I_2(t)$, ..., $I_{K_B}(t)$ - токи ветвей схемы A; $V'_1(t), V'_2(t), ..., V'_{K_B}(t)$ - напряжение ветвей, $I'_1(t), I'_2(t), ..., I'_{K_B}(t)$ - токи ветвей схемы A'.

Отметим, что в формулах (13.4) и (13.5) напряжения V и токи I принадлежат разным схемам, так что формулы не имеют какого-либо физического смысла "суммы энергий ветвей".

13.3. Схема присоединенной цепи

Две линейные схемы N и N' с независимыми параметрами называются взаимно присоединенными, если для них выполняются следующие три условия:

1) обе схемы имеют одинаковую топологию. Для управляемых источников мы рассматриваем в качестве управляющего напряжения напряжение на разомкнутой ветви, а в качестве управляющего тока - ток замкнутой накоротко ветви;

2) если ветви, не содержащие независимых источников, порождают матрицы полных проводимостей ветвей Y_b и $Y_b^{'}$, то $Y_b^t = Y_b^{'}$;

3) соответствующие независимые источники обеих схем одинаковы по своей природе, но необязательно одинаковы по значениям.

Приведенное определение присоединенных схем является практической рекомендацией, как построить схему, присоединенную по отношению к произвольной заданной схеме.

Рассмотрим определение коэффициентов чувствительности для резистивной схемы A (рис.13.4,*а*).

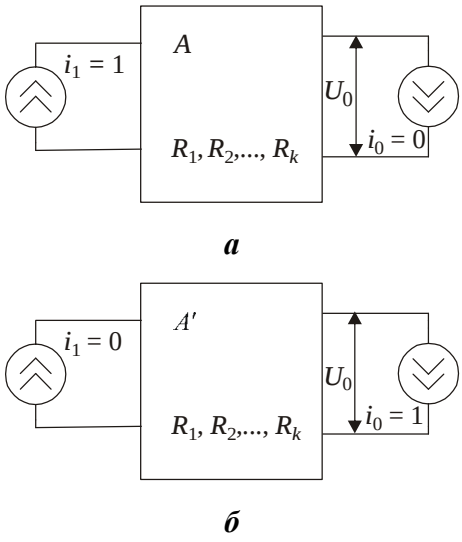


Рис.13.4. Общий вид схем: *а* - схема A; *б* - схема A'

На рис.13.4 к выходу схемы U_0 подключен дополнительный источник тока I_0 (его сопротивление $=\infty$), не влияющий на работу схемы, но обеспечивающий удобство дальнейших преобразований.

Рассмотрим вторую схему A', топологически идентичную первой (рис.13.4,*б*).

Компонентное уравнение для n-й ветви

$$I_n \cdot R_n = U_n. \qquad (13.6)$$

Выражение для приращений

$$I_n \Delta R_n + R_n \Delta i_n = \Delta U_n . \quad (13.7)$$

Для входной цепи i_i = const, так что $\Delta i_i = 0$, $\Delta i_0 = 0$.

Из теоремы Теллиджена следует, что

$$\sum_{n=1}^{K_B} (V_n + \Delta V_n) I'_n = 0; \quad \sum_{n=1}^{K_B} V'_n (I_n + \Delta I_n) = 0 .$$

Учитывая, что

$$\sum_{i=1}^{K_B} V_i(t) I_i(t) = 0 \quad \text{и} \quad \sum_{i=1}^{K_B} V'_i(t) I'_i(t) = 0 ,$$

получим:

$$\sum_{n=1}^{K_B} \Delta V_n I'_n = 0 ; \quad \sum_{n=1}^{K_B} V'_n \Delta I_n = 0 ;$$

$$\sum_{n=1}^{K_B} (\Delta V_n I'_n - V'_n \Delta I_n) = 0 \quad \text{или}$$

$$\Delta V_i I'_i - \Delta I_i V'_i + \Delta V_0 I'_0 - \Delta I_0 V_0 + \sum_{n=1}^{K} (\Delta V_n I'_n - V'_n \Delta I_n) = 0 .$$

Отсюда

$$\Delta V_i I'_i + \Delta V_0 I'_0 + \sum_{n=1}^{K} ((R_n I'_n - V'_n) \Delta I_n + (I_n I'_n) \Delta R_n) = 0. \quad (13.8)$$

Эту формулу можно упростить путем определенного выбора элементов цепи A'. Например, n-е элементы этой цепи являются резисторами с сопротивлением R_n, тогда $V'_n = R_n I'_n$. Поэтому в формуле (13.8) $R_n I'_n - V'_n = 0$ и она принимает вид

$$\Delta V_i I'_i + \Delta V_0 I'_0 + \sum_{n=1}^{K} I_n I'_n \Delta R_n = 0. \quad (13.9)$$

Представим i-й элемент цепи источником тока величиной $I'_n = 0$, а $I'_0 = 1$. В результате получим

$$\Delta V_0 I_0' = -\sum_{n=1}^{K} I_n I_n' \Delta R_n = 0. \qquad (13.10)$$

Так как варьируется только один резистор l, то $\Delta R_n = 0$ для всех $n \neq l$ и

$$\Delta V_0 = -I_l I_l' \Delta R_l .$$

Откуда можно получить

$$S_{R_l} = \frac{\Delta V_0}{\Delta R_l} = -I_l \cdot I_l', \qquad (13.11)$$

где S_{R_l} - чувствительность выходного напряжения V_0 по элементу R_l; I_l, I'_l - токи l-ных элементов исходной и присоединенной схем.

Достоинство метода в том, что для получения коэффициентов чувствительности по всем элементам l необходимо промоделировать две схемы - A и A'.

Недостаток метода - расчет проводится только для одного выхода.

13.4. Метод дифференцирования уравнений

Представим ММС в виде:

$$I(\varphi, E, p) = 0, \qquad (13.12)$$

где $\varphi = (\varphi_1, \varphi_2, ..., \varphi_n)^T$ - вектор-столбец узловых потенциалов; $E = (E_1, E_2, ..., E_n)$ - вектор-строка источников напряжения; $P = (p_1, p_2, ..., p_n)$ - вектор-строка параметров компонентов схемы.

Если φ* - решение уравнения (13.12), то справедлива система тождеств

$$I(\varphi^*, E, p) = 0. \qquad (13.13)$$

Продифференцировав (13.13) по параметрам p, получим:

$$\left(\frac{\delta I}{\delta \varphi^*}\right)\left(\frac{\delta \varphi^*}{\delta p}\right) + \frac{\delta I}{\delta p} = 0. \qquad (13.14)$$

Отсюда

$$\left(\frac{\delta\varphi^*}{\delta p}\right) = -\left(\frac{\delta I}{\delta\varphi^*}\right)^{-1}\frac{\delta I}{\delta p}. \qquad (13.15)$$

Матрица коэффициентов чувствительности узловых потенциалов по внешним параметрам E вычисляется аналогично формуле (13.16)

$$\left(\frac{\delta\varphi^*}{\delta E}\right) = -\left(\frac{\delta I}{\delta\varphi^*}\right)^{-1}\frac{\delta I}{\delta E}. \qquad (13.16)$$

Формулы (13.15) и (13.16) являются основой для расчета коэффициентов чувствительности методом дифференцирования уравнений.

Лекция 14

Параметрическая оптимизация электронных схем. Общие сведения

Конец? Нелепое словцо!
Чему конец? Что, собственно, случилось?
Раз нечто и ничто отождествилось,
То было ль вправду что-то налицо?
Зачем же созидать? Один ответ:
Чтоб созданное все сводить на нет.

И.В. Гёте. Фауст

Процесс параметрической оптимизации электронных схем, как известно, заключается в таком выборе структуры и подборе параметров компонентов, который обеспечивает функционирование схем в соответствии с требованиями технического задания. С точки зрения математиков, этот процесс можно рассматривать как сведéние к минимуму некоторой меры ошибок. ***Мера ошибок известна как критерий оптимизации (критерий качества, функция качества, целевая функция).***

Математически задачу оптимизации можно сформулировать как задачу поиска экстремума целевой функции $\Phi(P)$, где P - вектор изменяемых параметров при наличии ограничений

$$\Phi(P) \geq 0, \quad i = 1, 2, \ldots, m, \qquad (14.1)$$

где m - количество ограничений. Эта система ограничений задает так называемую ***область работоспособности***.

В общем случае целевая функция может быть как векторной, так и скалярной.

Под областью работоспособности понимают область в пространстве параметров, в которой выполняются условия работоспособности рассматриваемой схемы. При оптимизации электронных схем в качестве изменяемых параметров P обычно выступают параметры элементов (транзисторов, диодов, резисторов и т.д.). Неравенства (14.1) задают условия работоспособности схем и содержат требования на выходные параметры разрабатываемой схемы и ограничения на параметры элементов схем, обусловленные возможностью их физической реализации. В качестве целевой функции используется один из выходных параметров или их совокупность. Отметим, что наличие ограничений

существенно усложняет задачу оптимизации; точкой экстремума часто оказывается некоторая граничная точка области работоспособности.

Различают ***детерминированные и статистические критерии оптимизации.*** Эти критерии могут быть обобщенными. При использовании детерминированного критерия оптимизации в качестве целевой функции используют один из выходных параметров. В качестве детерминированного обобщенного критерия оптимизации может быть использована комбинация нескольких выходных параметров электронных схем, взятых с соответствующими весовыми коэффициентами, характеризующими важность соответствующих выходных параметров для конкретного потребителя.

В качестве статистического критерия оптимизации может быть взято отношение в процентах числа годных схем, попавших в область работоспособности, к общему числу изготовленных схем, - процент выхода годных.

Далее будут рассматриваться только методы параметрической оптимизации.

Основные этапы решения задачи оптимизации. Решение любой задачи параметрической оптимизации начинается с выбора критерия оптимизации. При выборе критерия оптимизации необходимо пользоваться следующими положениями.

1. Характер выбранной целевой функции влияет на выбор эффективного метода поиска ее экстремума.

2. Целевая функция должна быть такой, чтобы оптимизация выполнялась за приемлемое время.

3. Целевая функция и метод оптимизации должны быть общими для широкого класса задач.

Затем необходимо сформулировать систему ограничений, в пределах которой ищется оптимальное решение. На практике число ограничений примерно 50 - 70, так как к ограничениям на электрические характеристики добавляются ограничения, связанные с физической реализуемостью компонентов.

После этого можно приступить к поиску оптимума полученной целевой функции. Поиск оптимального решения может быть сформулирован как общая задача нелинейного программирования.

Основные этапы поиска оптимума следующие.

1. Выбор начального приближения в пространстве изменяемых параметров P. Исходная точка P_0 может находиться как вне, так и внутри области работоспособности. Для сокращения времени поиска желателен второй вариант. Для этого следует предварительно провести ручной расчет P_0.

Во многих случаях целью оптимизации является улучшение работы уже функционирующей схемы. Тогда внутренние параметры элементов этой схемы часто являются хорошим приближением для дальнейшей оптимизации.

2. Выбор метода оптимизации с учетом вида целевой функции.

3. Определение величины шага поиска.

4. Вычисление целевой функции на шаге поиска.

5. Определение окончания этапа поиска.

При этом проверяется попадание экстремума целевой функции в заданную ε - окрестность P^* пространства изменяемых параметров.

Рассмотрим некоторые методы поиска экстремума целевых скалярных функций конечного числа переменных. Будем считать, что целевая функция $\Phi(P)$ является унимодальной.

Определение унимодальности функции. Предположим, что f - действительная функция, определенная на отрезке [0,1]. Предположим далее, что имеется *единственное значение* $\underline{x}$, такое, что $f(\hat{x})$ - максимум $f(x)$ на отрезке [0,1], и что $f(x)$ строго возрастает для $x \le \hat{x}$ и строго убывает для $\hat{x} \le x$, т.е. функция имеет один экстремум. Такая функция называется *унимодальной*. В противном случае функцию называют *мультимодальной*.

Для графика унимодальной функции имеются три возможные формы (рис.14.1).

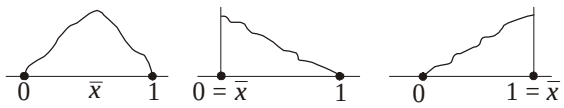


Рис.14.1. Возможные графики унимодальной функции

По отношению к экстремумам мультимодальной функции применяют термины "абсолютный", "относительный", "локальный" и "глобальный".

Заметим, что унимодальная функция не обязана быть гладкой или даже непрерывной. Все рассматриваемые далее алгоритмы поиска экстремума дают лишь локально оптимальные решения. Если требуется определить глобальный минимум, то можно, например, использовать несколько начальных точек поиска P_0.

Классификация методов оптимизации. Методы оптимизации можно классифицировать по следующим признакам.

1. ***По числу переменных:***

- одномерная оптимизация;
- многомерная оптимизация.

Заметим, что большинство методов многомерной оптимизации в конечном итоге сводится к задаче одномерной оптимизации. Отметим также, что при минимизации функции большого числа переменных часто невозможно получить численное решение. Поэтому на первых этапах оптимизации важно построить простейшую модель, учитывающую лишь основные, определяющие параметры. Эти параметры можно выделить, анализируя коэффициенты матрицы чувствительности (см. лекцию 13).

2. ***По использованию в процессе оптимизации информации о целевой функции и ее производных:***

- ***методы нулевого порядка (прямые методы).*** В этих методах используется только знание значений целевой функции;
- ***методы первого порядка (градиентные методы).*** В этих методах, помимо информации о целевой функции, используется информация о направлении скорейшего изменения данной целевой функции, т.е. значение градиента;
- ***методы второго порядка.*** Эти методы дополнительно требуют информацию о вторых частных производных целевой функции.

3. ***По выбору направления движения:***

- алгоритмы поиска точки минимума можно разделить на ***детерминированные***, если направление движения от точки P_i к точке P_{i+1} на шаге перехода выбирается однозначно по доступной в точке P_i информации. Если при переходе применяется какой-либо случайный механизм, то алгоритм поиска называется ***случайным*** поиском минимума.

4. ***По использованию информации, полученной на предыдущих шагах оптимизации***:

- алгоритмы поиска *экстремума* разделяются на ***алгоритмы с памятью*** и ***алгоритмы без памяти***. В первой группе алгоритмов на очередном шаге используются данные предыдущих шагов, во второй - информация только о данной точке.

При оценке качества алгоритма учитываются такие показатели, как скорость сходимости к решению, количество используемых машинных операций, объем требуемой оперативной памяти, чувствительность к ошибкам округления и т.п.

С математической точки зрения задачу оптимизации можно сформулировать следующим образом: найти множество абсцисс x_1, x_2, ..., x_k, в которых вычисляется функция, такое, что оптимальное значение f лежит при некотором i в интервале $x_{i-1} \le x \le x_{i+1}$ (этот интервал называется интервалом неопределенности или интервалом локализации).

Алгоритм выбора абсцисс x_i (i = 1, …, k) называется планом поиска. Если известно только то, что f - унимодальная функция, то возникает вопрос о том, какова оптимальная стратегия для нахождения x. Можно считать, что при заданном количестве вычислений функции оптимальным планом поиска будет тот, который приводит к наименьшему интервалу неопределенности.

14.1. Методы одномерной оптимизации

Рассмотрим несколько широко известных методов одномерной оптимизации. Пусть имеется унимодальная целевая функция Ф(p), изображенная на рис.14.2, где p - скаляр.

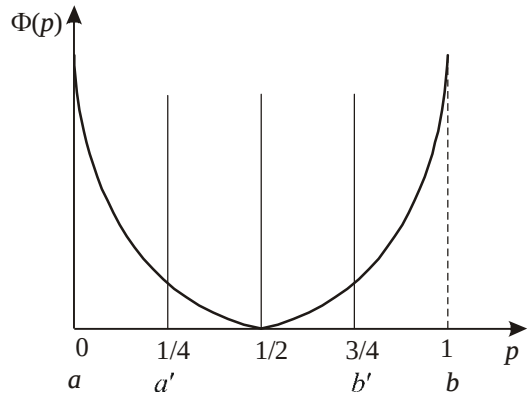


Рис.14.2. Иллюстрация метода дихотомии

Задача оптимизации состоит в определении минимума этой функции на интервале [a, b] допустимого изменения переменной p.

Определим координату точки p относительно интервала [a, b] как число, равное [p – a]/[b – a]. Таким образом, a имеет по отношению к интервалу [a, b] координату 0, b - координату 1, а средняя точка - координату 1/2.

Иными словами, представим диапазон изменения параметра p единичным отрезком. На данном отрезке зададим N значений переменной, включая крайние точки 0, 1. Далее вычислим значения целевой функции в этих точках и определим точку N_0, в которой значение целевой функции минимально. В силу унимодальности вычисляемой функции экстремум целевой функции находится в интервале локализации между точками, ближайшими справа и слева к N_0.

Отметим, что в общем случае число точек N и расстояние между ними может быть произвольным. Повторим описанную процедуру поиска экстремума для нового интервала, в котором находится минимум целевой, и будем действовать так до тех пор, пока на очередном шаге интервал не станет меньше заданного.

Предлагаемые далее методы поиска экстремума скалярной различаются планами поиска.

Примечание. Будем считать, что экстремум является точкой минимума.

1. **Метод дихотомии (деления отрезка пополам).** Разделим полученный в результате нормирования отрезок на оси абсцисс на четыре равные части и вычислим значение целевой функции в пяти точках: 0, 1/4, 1/2, 3/4, 1. Затем выберем минимальное из пяти найденных значений. Очевидно, что экстремум функции лежит в границах нормированного участка [1/4, 3/4] (соответственно участка [a', b'] изменения переменной p), прилегающего с двух сторон к точке найденного минимума. Следовательно, интервал поиска сужается в два раза. Если минимум находится в точке 0 (a) или 1 (b), то интервал сужается в четыре раза.

Полученный интервал поиска вновь разбиваем на четыре равные части и повторяем вычисления. Поскольку значения целевой функции в трех точках нового интервала (двух крайних и центральной) были вычислены на предыдущем этапе, следующий этап требует вычисления целевой функции в двух новых точках. Описанный процесс останавливаем, когда интервал неопределенности становится меньше заданного.

2. **Метод Фибоначчи.** В основу метода положен ряд чисел, называемых числами Фибоначчи. Это ряд $F_0 = F_1 = 1$, $F_2 = 2$, $F_3 = 3$, $F_4 = 5$, $F_5 = 8$,…, $F_k = F_{k-1} + F_{k-2}$. Предлагаемая стратегия последовательного поиска экстремума называется поиском Фибоначчи, поскольку она тесно связана с этими числами. При оптимальной стратегии поиска выбираем два значения переменной - $p_{k-1} = F_{k-2}/F_k$ и $p_k = F_{k-1}/F_k$ и проводим вычисление целевой функции в четырех точках. Какой бы из интервалов - [0, p_k] или [p_{k-1}, 1] не стал суженным интервалом неопределенности, в нем находится "унаследованная" точка, т.е. точка, значение целевой функции в которой уже вычислено и будет использовано на последующем шаге. Она будет иметь по отношению к новому интервалу одну из двух следующих координат: F_{k-3}/F_{k-1} или F_{k-2}/F_{k-1}.

На втором шаге вычисление целевой функции проводим только для одной новой переменной p_{k-2}. Используя $\Phi(p_{k-2})$ и значение функции, унаследованное от прежнего интервала, сокращаем интервал неопределенности и передаем в наследство следующему шагу одно значение функции.

На последнем шаге мы придем к некоторому интервалу неопределенности [a, b], причем средняя точка будет унаследованной от предыдущего шага.

Тогда в качестве p_1 выбирается точка с относительной координатой 1/2 + ε, и окончательным интервалом неопределенности будет либо [0, 1/2 + ε], либо [1/2, 1] относительно [a, b].

На первом шаге длина интервала неопределенности уменьшилась с 1 до F_{k-1}/F_k. На последующих шагах уменьшение длин интервалов выражается числами

$$F_{k-2}/F_{k-1},\ F_{k-3}/F_{k-2},\ \ldots,\ F_2/F_3,\ F_1/F_2(1 + 2\varepsilon).$$

Таким образом, длина окончательного интервала неопределенности равна $(1 + 2\varepsilon)/F_k$. Пренебрегая ε, заметим, что асимптотически F_{k-1}/F_k равно r при $k \to \infty$. Здесь

$$r = (\sqrt{5} - 1)/2 \approx 0{,}6180. \quad (14.2)$$

Заметим, что асимптотически для больших k каждый шаг поиска Фибоначчи сужает интервал неопределенности с коэффициентом ≈ 0,6180, т.е. медленнее, чем в методе деления отрезка пополам, однако на каждом шаге, начиная со второго, требуется вычислять целевую функцию только в одной точке, а не в двух, как требует метод деления отрезка пополам.

Пример. Зададим максимальное число Фибоначчи, равное 55. Разделим единичный интервал поиска минимума [0, 1] на три части точками 21/55 и 34/55. Предположим, что значение Ф(34/55) - минимальное из четырех, вычисленных в указанных точках. Следовательно, интервал неопределенности лежит в пределах [21/55, 1], а его длина - 34/55.

Перейдем ко второму шагу. Будем считать полученный интервал единичным и зададим максимальное число Фибоначчи, равное 34. Разделим единичный интервал поиска минимума [0, 1] ([21/55, 1] исходного интервала) на три части точками 13/34 и 21/34. Очевидно, что точка 21/34 нового интервала неопределенности соответствует точке 34/55 исходного.

Следовательно, в новом интервале нужно вычислять целевую функцию только для одной точки 13/34. Предположим, что значение Ф(13/34) - минимальное из четырех, вычисленных в указанных точках интервала поиска.

Таким образом, новый интервал неопределенности лежит в пределах [0, 21/34] ([21/55, 34/55] исходного интервала), а его длина 13/55. Вновь, считая полученный интервал единичным, зададим максимальное число Фибоначчи, равным 21, и продолжим итерационный процесс.

3. **Метод золотого сечения.** Известно, что "золотое сечение" отрезка $[a, b]$ - это такое сечение, когда отношение r большей части $[a, c]$ отрезка к длине всего отрезка $[a, b]$ равно отношению меньшей части $[c, b]$ к большей части $[a, c]$.

Нетрудно показать, что r выражается формулой (14.2) для больших номеров k ряда чисел Фибоначчи. Следовательно, при больших значениях k метод "золотого сечения" по сути аналогичен методу Фибоначчи. Но, в отличие от метода Фибоначчи, здесь не нужно задавать число k до начала поиска.

Действительно, при больших значениях k координаты точек p_{k-1} и p_k в методе Фибоначчи близки соответственно к $1 - r \approx 0{,}3820$ и $r \approx 0{,}6180$, и выбор этих значений близок к оптимальной стратегии. Таким образом, единичный интервал поиска минимума [0, 1] методом "золотого сечения" делим на три части точками $1 - r$ и r. Затем, аналогично методу Фибоначчи, находим минимум целевой функции $\Phi(p_i)$ в четырех точках. Предположим для определенности, что $\Phi(0{,}3820)$ - наименьшее из рассчитанных значений. Тогда, как мы знаем, решение $\hat{x}$ находится в интервале неопределенности [0, 0,6180]. Полученный интервал также делим на части в соотношении "золотого сечения". Следовательно, нужно вычислять $\Phi(p_i)$ в точках $0{,}3820 \cdot 0{,}6180$ и $0{,}6180 \cdot 0{,}6180$. Но, поскольку $0{,}6180 \cdot 0{,}6180 \approx 0{,}3820 \approx p_{k-1}$, то в этой точке значение $\Phi(p_i)$ уже известно.

Таким образом, на каждом шаге, начиная со второго, требуется вычисление функции лишь в одной точке, и каждый шаг уменьшает длину интервала неопределенности с коэффициентом 0,6180.

4. **Метод удвоения шага.** Идея метода заключается в поиске направления убывания функции и движении в этом направлении с возрастающим шагом при удачном поиске. Предлагается один из вариантов алгоритма поиска минимума целевой функции.

1. Выбираем начальную координату P_0 целевой функции $\Phi(p)$, минимальную величину шага h_0 и направление поиска.
2. Вычисляем значение функции в точке P_0.
3. Делаем шаг в выбранном направлении и вычисляем значение целевой функции в следующей точке $P_0 + h_0$.
4. Если значение функции меньше, чем на предыдущем шаге, то делаем следующий шаг в этом же направлении, увеличив его вдвое ($h = 2h_0$). Величину каждого последующего шага удваиваем, пока на очередном шаге целевая функция не увеличится. После того, как значение целевой функции станет больше, чем на предыдущем шаге, меняем направление поиска и начинаем двигаться в этом направлении с начальным значением шага h_0.
5. Прекращаем поиск после того, как интервал неопределенности сократится до значения $2h_0$.

14.1.1. Интерполяция целевой функции

В практике применения целевых функций часто оказывается, что в окрестности решения целевую функцию можно аппроксимировать простыми формами (например, квадратичными или кубическими).

Рассмотрим пример квадратичной интерполяции целевой функции $\Phi(p)$. Пусть за n шагов интервал поиска оптимума сужается до $[p_{n1}, p_{n3}]$. Считаем, что в полученном интервале целевая функция достаточно точно описывается квадратным уравнением (рис.14.3).

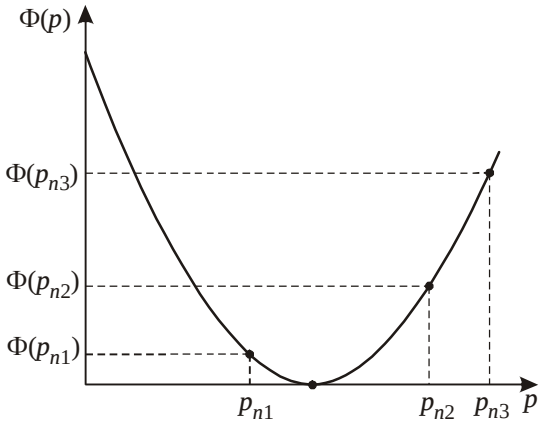


Рис.14.3. Пример квадратичной интерполяции целевой функции

Составим уравнения для крайних точек p_{n1} и p_{n3} интервала неопределенности:

$$ap_{n1}^2 + bp_{n1} + c = \Phi(p_{n1}); \quad (14.3)$$

$$ap_{n3}^2 + bp_{n3} + c = \Phi(p_{n3}). \quad (14.4)$$

В этих уравнениях неизвестными являются коэффициенты a, b и c. Для их определения нужно составить еще одно, третье уравнение. В этих целях выберем случайную точку p_{n2} внутри интервала и вычислим значение целевой функции в данной точке. Тогда к уравнениям (14.3), (14.4) можно добавить еще одно уравнение

$$ap_{n2}^2 + bp_{n2} + c = \Phi(p_{n2}). \quad (14.5)$$

Решая полученную систему уравнений (14.3) - (14.5), можно определить коэффициенты a, b, c.

Как известно, минимум целевой функции $\Phi(p_x) = ap_x^2 + bp_x + c$ находится в точке, где ее производная обращается в нуль. В этом случае координата минимума $p_x = -b/(2a)$.

Первый и второй подходы, как было показано ранее, обычно приводят к решению систем линейных алгебраических уравнений. Третий подход также часто требует решения таких систем.

14.2. Методы многомерной оптимизации

Минимизация целевой функции осуществляется путем применения процедур получения последовательности векторов $P_1, P_2, P_3, \ldots, P_i$ в пространстве параметров, таких, что

$$\Phi(P_1) > \Phi(P_2) > \Phi(P_3) > \ldots. > \Phi(P_1). \quad (14.6)$$

В дальнейшем будем рассматривать задачу минимизации целевой функции как движение, и называть движение ***спуском.*** Любой метод получения начальных значений векторов, соответствующих условию (15.1), назовем ***"методом спуска".***

Методы многомерной оптимизации различаются либо выбором направления, в котором осуществляется поиск экстремума, либо способом движения вдоль этого направления.

14.3. Методы нулевого порядка (прямые методы оптимизации)

14.3.1. Метод покоординатного спуска

Простейшим методом нулевого порядка является ***метод покоординатного спуска***, где в качестве векторов S_j, определяющих направление спуска, последовательно выбирают единичные координатные векторы. Основная идея метода состоит в том, что на каждом шаге поиска все переменные (внутренние и внешние параметры), кроме одной, фиксируются как постоянные. Затем минимизируется целевая функция $\Phi(P)$ относительно одного изменяемого параметра. Движение в выбранном направлении определяется формулой

$$P_{j+1} = P_j + \Delta P_{j+1}. \quad (14.7)$$

Здесь $\Delta P_{j+1} = \beta_{j+1} \cdot S_{j+1}$. Приращение β_{j+1} в заданном направлении S_{j+1} выбирается таким образом, чтобы целевая функция $\Phi(P)$ уменьшалась, т.е. осуществлялся спуск. В частном случае алгоритм требует спуска до точки минимума в заданном направлении. Геометрическая интерпретация данного метода показана на рис.14.4, где унимодальная целевая

функция Ф(P) представлена графически в виде проекций линий ее равного уровня на плоскость переменных p_1, p_2. На этом рисунке показана траектория движения от начальной точки поиска $P_0 = (p_{1,0}, p_{2,0})$ к минимуму. При этом на каждом шаге поиска спуск по ортогональным направлениям ведется до тех пор, пока в данном направлении не будет достигнут локальный минимум.

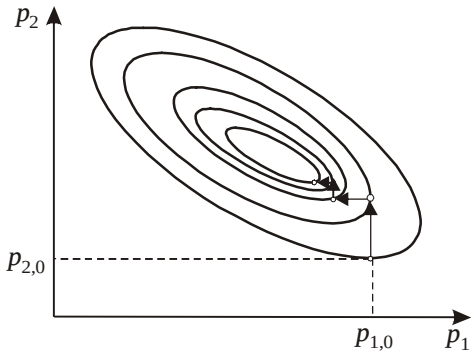


Рис.14.4. Графическая иллюстрация метода покоординатного спуска

Существенным недостатком рассмотренного метода является ***"зигзагообразный"*** характер движения к точке минимума по многомерной "плоскости" переменных, что значительно замедляет его сходимость.

Отмеченный недостаток можно устранить, применяя ***методы случайного поиска,*** иначе - ***вероятностные методы***. Их преимущества проявляются прежде всего в случае большого числа параметров, по которым проводится оптимизация.

14.3.2. Методы случайного поиска

Рассмотрим один из алгоритмов, реализующий случайный поиск экстремума. Предлагаемый алгоритм носит название алгоритма ***"наилучшей пробы".***

Э т а п 1. Выберем две случайные точки в пространстве параметров изменяемых компонентов, свяжем их прямой и найдем локальный экстремум на полученной прямой одним из известных методов одномерной минимизации.

Э т а п 2. Повторим операцию для двух следующих случайных точек.

Э т а п 3. Будем искать локальный минимум на прямой, проходящей через точки локальных экстремумов, полученных ранее.

Э т а п 4. Выполним этап 1 и перейдем к этапу 3.

Следовательно, направление спуска, определяемое на этапе 3, зависит от предыдущих направлений. Такие направления называются ***сопряженными***. Дадим следующее определение.

Определение. Для данной симметричной матрицы A порядка m направления $S_1, S_2, \ldots, S_r$ ($r < m$) называются сопряженными, если S_i линейно независимы и

$$S_i^T A S_j = 0 \text{ для всех } i \neq j. \quad (14.8)$$

Основным свойством сопряженных направлений, позволяющим эффективно использовать их для решения задачи минимизации, является следующее.

Пусть задана квадратичная функция вида

$$\Phi(P) = B^m P + 0{,}5 P^m A P, \quad (14.9)$$

где A - положительно определенная симметричная матрица и m - число сопряженных направлений $S_1, S_2, \ldots, S_m$. Проведя минимизацию $\Phi(P)$ по каждому из направлений $S_1, S_2, \ldots, S_m$ (как положительному, так и отрицательному), найдем минимум квадратичной функции, зависящий от m параметров. Минимум может быть найден не более чем за m итераций.

Отметим, что различные методы минимизации, использующие сопряженные направления, отличаются в основном способом определения этих направлений. Опишем один из таких способов для произвольных квадратичных функций (модификация ранее рассмотренного алгоритма). Этот способ позволяет по данным $r < m$ сопряженным направлениям построить новое сопряженное направление.

Пусть $S_1, S_2, \ldots, S_r$, $r < m$ - сопряженные направления. Предположим, что $P' = (p'_1, p'_2, \ldots, p'_m)$, $P'' = (p''_1, p''_2, \ldots, p''_m)$ получены путем минимизации $\Phi(P)$ по каждому из направлений $S_1, S_2, \ldots, S_r$ из различных начальных точек, и $\Phi(P') > \Phi(P'')$. Тогда направления $S_1, S_2, \ldots, S_r, S_{r+1}$, где $S_{r+1} = P'' - P'$, являются сопряженными.

Точки P'', P' могут быть получены следующим образом.

Пусть $P_1 = (p_{1,1}, p_{2,1}, \ldots, p_{m,1})$ - произвольная начальная точка. Допустим, что по этой точке, минимизируя $\Phi(P)$ по каждому из направлений $S_1, S_2, \ldots, S_r$, мы определяем точку P'. Если эта точка не является оптимальной, то всегда можно найти такую точку $P_2 = (p_{1,2}, p_{2,2}, \ldots, p_{m,2})$, что $\Phi(P') > \Phi(P_2)$. Выполнив для точки P_2 то же, что и для точки P_1, мы получим точку P'', причем, так как $\Phi(P_2) > \Phi(P'')$, то $\Phi(P') > \Phi(P'')$.

Процедура построения сопряженного направления для двумерного случая иллюстрируется рис.14.5.

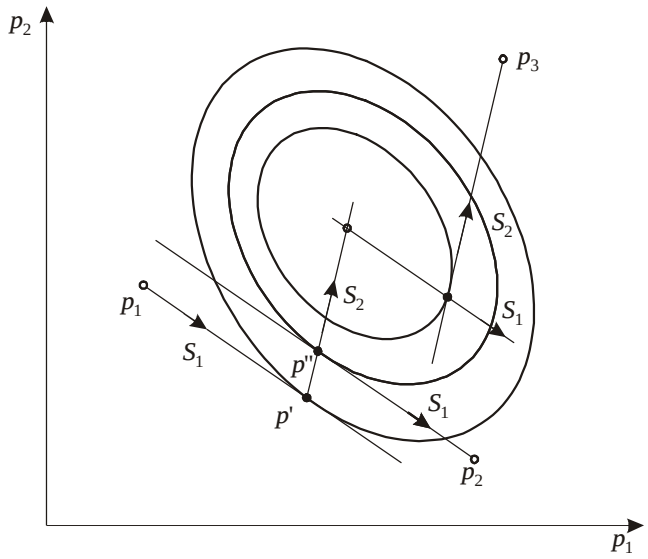


Рис.14.5. Графическая иллюстрация метода построения сопряженных направлений (двумерный случай)

Здесь P_1 и S_1 - заданные исходная точка и первое направление. Точка P' получается путем минимизации функции вдоль прямой, проходящей через точку P_1 в направлении S_1, P_2 - вторая заданная начальная точка P'' получается путем минимизации функции вдоль прямой, проходящей через точку P_2 в том же направлении S_1. Сопряженное направление S_2 функции проходит через точки P', P''. Минимум $\Phi(P)$ может быть найден из произвольной точки P_3 за две минимизации функции вдоль полученных сопряженных направлений S_1 и S_2.

Рассмотренный метод построения дополнительного сопряженного направления лежит в основе ***модификации следующего алгоритма "наилучшей пробы"***.

1. Выбираем начальную точку P_0^0 и произвольное направление $S_1 \neq 0$ (это направление может совпадать, например, с одним из координатных направлений). Положим $k = 1$.
2. Определяем точку P_0^k путем минимизации вдоль прямой, проходящей через точку P_0^k, в направлении S_k.
3. Определяем точку P_1^k такую, что $\Phi(P_1^k) < \Phi(P_0^k)$, используя одну итерацию метода покоординатного спуска. Если такую точку найти не удается, то нужно остановиться, иначе - перейти к шагу 4.
4. Задаем $i = 1$. Определяем точку P_{i+1}^k путем минимизации функции вдоль прямой, проходящей через точку P_i^k в направлении S_i.

5. Задаем $i = i + 1$. Если $i \le k$, то перейти к шагу 4, иначе - к шагу 6.

6. Задаем $S_{k+1} = P_{k+1}^k - P_0^k$ и $k = k + 1$. Переходим к шагу 2.

Данный алгоритм прекращает работу, когда ни по одной из координат не удается получить уменьшения функции $\Phi(P)$. Это, очевидно, произойдет в некоторой точке P_0^k, в которой градиент минимизируемой функции равен нулю: $\nabla\Phi(P_0^k) = 0$, т.е. выполняется необходимое условие оптимума.

Для квадратичной функции остановка алгоритма произойдет не позже, чем через m итераций, когда будут определены m сопряженных направлений.

В случае минимизации функции общего вида с помощью данного алгоритма вычисление сопряженных направлений невозможно. Однако можно представить себе, что вычисляемые направления являются сопряженными для некоторой квадратичной функции, которая аппроксимирует $\Phi(P)$. При приближении к экстремуму такая аппроксимация становится, как правило, все более точной, что устраняет зигзагообразное движение, характерное для метода покоординатного спуска. Следует отметить, что при минимизации функции общего вида данный алгоритм обычно повторяют через каждые m итераций.

14.4. Градиентные методы оптимизации

14.4.1. Методы первого порядка

Градиентные методы оптимизации основаны на разложении целевой функции $\Phi(P)$ в окрестности точки P_i в ряд Тейлора и пренебрежении членами разложения второго и более высоких порядков. Второй член разложения, учитываемый в методах, представляет собой вектор-градиент целевой функции $\Phi(P)$, где $P = P_i$. Полученное направление является направлением максимального убывания функции.

Одним из наиболее известных и простейших методов первого порядка является метод наискорейшего спуска Коши. В этом методе в качестве вектора S_j, определяющего направление, в котором уменьшается целевая функция, выбирается антиградиент этой функции. Новые значения вектора переменных вычисляются по формуле

$$P_{j+1} = P_j - \alpha_j \nabla\Phi(P_j), \quad (14.10)$$

где α - величина шага, сделанного в направлении антиградиента.

14.4.2. Алгоритм метода наискорейшего спуска Коши

Ш а г 1. Задаем начальное значение вектора переменных P_0.

Ш а г 2. Находим направление антиградиента минимизируемой функции в заданной точке.

Ш а г 3. Находим точку локального экстремума P_1 в полученном направлении.

Ш а г 4. Задаем $P_0 = P_1$.

Ш а г 5. Переходим к шагу 1.

Данный алгоритм прекращает работу, когда $\nabla\Phi(P_1) = 0$.

Среди методов, использующих градиент для выбора направления поиска, этот метод наиболее простой. Однако его существенным недостатком является чрезвычайно медленная сходимость при плохой обусловленности матрицы вторых частных производных функции, т.е. если отношение наибольшего собственного значения матрицы к наименьшему в некоторой точке минимума велико. Траектория движения в окрестности такой точки при этом имеет зигзагообразный характер (рис.14.6), что требует иногда проведения сотен итераций для достижения минимума с приемлемой точностью. Поэтому в данном случае метод наискорейшего спуска практически не применяют.

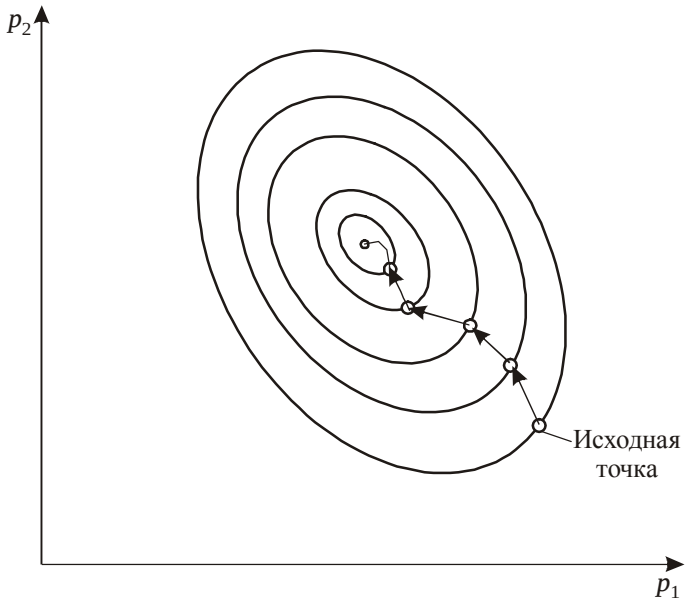


Рис.14.6. Графическая иллюстрация метода наискорейшего спуска Коши

Наиболее перспективными методами первого порядка являются методы, основанные на использовании сопряженных направлений. Как уже было отмечено, минимизация квадратичной функции по сопряженным направлениям позволяет получить ее минимум не более чем за m шагов. Отличительной особенностью этих методов является использование информации о предыдущих направлениях поиска при определении очередного на-

правления. Рассмотрим два наиболее распространенных метода, отличающихся способом определения сопряженных направлений.

14.4.3. Метод Флетчера – Ривса

Ш а г 1. Выбираем исходную точку P_0 и вычисляем $S_0 = -\nabla\Phi(P_0)$. Полагаем $k = 0$.

Ш а г 2. Определяем точку P_{k+1} путем минимизации функции $\Phi(P)$ в направлении S_k.

Ш а г 3. Вычисляем $\nabla\Phi(P_{k+1})$. Если $\nabla\Phi(P_{k+1}) = 0$, то прекращаем поиск, иначе переходим к шагу 4.

Ш а г 4. Вычисляем новое направление по формуле

$$S_{k+1} = -\nabla\Phi(P_{k+1}) + \frac{\left\|\Phi(P_{k+1})\right\|^2}{\left\|\nabla\Phi(P_k)\right\|^2} \cdot S_k, \qquad (14.11)$$

где $\|\nabla\Phi\|$ - норма вектора $\nabla\Phi$. Задаем $k = k + 1$ и переходим к шагу 1.

Таким образом, очередное направление представляет собой линейную комбинацию градиента и предыдущего направления. Это позволяет избежать зигзагообразного движения, характерного для метода наискорейшего спуска, и минимизировать квадратичную функцию за m или менее итераций.

Для функции общего вида алгоритм рекомендуется восстанавливать через каждые $k \geq m + 1$ итераций, т.е. выбирать $S_{k+1} = -\nabla\Phi(P_{k+1})$, а не по формуле (14.11).

Рассмотрение методов первого порядка закончим описанием наиболее эффективного, по мнению ряда исследователей, алгоритма минимизации, предложенного Флетчером и Пауэллом. Алгоритм (иногда его называют алгоритмом "с переменной метрикой") широко используется и обладает хорошей вычислительной устойчивостью. Недостатком алгоритма является необходимость хранения в памяти компьютера матрицы размером $m \times m$, что может привести к трудностям его использования на машинах с малым объемом оперативной памяти для решения задач большой размерности.

14.4.4. Метод Флетчера – Пауэлла

Ш а г 1. Выбираем исходную точку P_0 и вычисляем $S_0 = -\nabla\Phi(P_0)$. Задаем $H^0 = I$ (I - единичная матрица размером $m \times m$) и $k = 0$.

Ш а г 2. Определяем точку P_{k+1} путем минимизации функции $\Phi(P)$ в направлении S_k.

Ш а г 3. Вычисляем $\nabla\Phi(P_{k+1})$. Если $\nabla\Phi(P_{k+1}) = 0$, то прекращаем поиск, иначе переходим к шагу 4.

Ш а г 4. Вычисляем

$$H^{k+1} = H^k + \frac{\Delta P_k (\Delta P_k)^T}{(\Delta P_k)^\tau d_k} - \frac{H^k d_k (d_k)^T H^k}{(d_k)^\tau H^k d_k}, \quad (14.12)$$

где $\Delta P_k = P_{k+1} - P_k$; $d_k = \nabla\Phi(P_{k+1}) - \nabla\Phi(P_k)$.

Формула (14.12) приближенно определяет матрицу вторых частных производных целевой функции (матрицу Гессе).

Ш а г 5. Определяем новое направление по формуле

$$S_{k+1} = -H^{k+1}\nabla\Phi(P_{k+1}). \quad (14.13)$$

Задаем $k = k + 1$ и переходим к шагу 2.

При минимизации положительно определенной квадратичной формы (14.9) направления $S_1, S_2, ..., S_{m-1}$ оказываются сопряженными относительно матрицы A, что обеспечивает сходимость метода в данном случае за m итераций. Кроме того, в этом алгоритме происходит обращение матрицы вторых частных производных в точке минимума , т.е. $H^m = A^{-1}$.

Использование метода Флетчера - Пауэлла для минимизации квадратичной функции приводит к построению точно такой же последовательности векторов P_j и S_j, как и в методе Флетчера - Ривса. Однако в более общем случае метод переменной метрики быстрее. Это можно объяснить тем, что, в отличие от метода Флетчера - Ривса, в методе Флетчера - Пауэлла при определении очередного направления учитывается не одно, а все предыдущие направления. Заметим, что в формуле (14.12) используется информация о предыдущих итерациях для определения неявным образом второй производной целевой функции.

Таким образом, вышеописанный метод можно рассматривать как метод второго порядка.

14.5. Методы второго порядка

Рассмотренный ранее метод наискорейшего спуска основан на линейной аппроксимации $\Phi_L(P)$ минимизируемой функцией $\Phi(P)$ в точке P_j:

$$\Phi_L(P) = \Phi(P_j) + \nabla\Phi(P_j)(P - P_j). \quad (14.14)$$

Данная аппроксимация является довольно точной в окрестности точки P_j, поэтому движение в направлении антиградиента локально является наиболее выгодным для минимизации $\Phi(P)$.

Методы второго порядка также основаны на разложении минимизируемой функции в ряд Тейлора, однако они используют на один член разложения больше. Типичным представителем методов второго порядка является метод Ньютона. Идея данного метода основана на возможности аналитического вычисления точки минимума квадратичной функции. При этом необходимые формулы получают следующим образом.

Разлагая $\Phi(P)$ в точке P_j в ряд Тейлора, получаем квадратичное приближение к $\Phi(P)$ вида

$$\Phi_0(P) = \Phi(P_j) + \nabla\Phi(P_j)(P - P_j) + 0{,}5(P - P_j)^T H(P_j)(P - P_j), \quad (14.15)$$

где $H(P_j)$ - матрица Гессе вторых частных производных $\Phi(P)$ в точке P_j.

Учитывая, что в точке минимума P^* функции $\Phi_Q(P)$ выполняется неравенство $\nabla\Phi_Q(P^*) = 0$, можно найти следующую формулу после дифференцирования (14.15):

$$P^* = P_j - H^{-1}(P_j)\nabla\Phi(P_j). \quad (14.16)$$

Соотношение (14.16) определяет положение точки минимума функции (14.15), которая является квадратичной аппроксимацией $\Phi(P)$. Таким образом, если минимизируемая функция является положительно определенной квадратичной формой, то формула (14.16) позволяет найти ее минимум за одну итерацию.

Для функции общего вида на основании формулы (14.16) можно записать следующую итерационную формулу Ньютона:

$$P_{j+1} = P_j - \alpha_j H^{-1}(P_j)\nabla\Phi(P_j), \quad (14.17)$$

где $\alpha_j \le 1$, т.е. формулу модифицированного метода Ньютона; иными словами, в данном методе в качестве S_j, определяющего направление поиска, выбирается вектор $-H^{-1}(P_j)\nabla\Phi(P_j)$. Формула (14.17) позволяет получать последовательность минимумов аппроксимирующих функций. Если минимизируемая функция выпуклая, то (14.17) гарантирует ее монотонное убывание от итерации к итерации.

Метод Ньютона обладает квадратичной скоростью сходимости, т.е.

$$\left|P_{j+1} - P^*\right| \le \gamma \left|P_j - P^*\right|^2,$$

где P^* - точка оптимума, а коэффициент γ зависит от вида минимизируемой функции. Однако для невыпуклых функций он сходится не из любых начальных точек. Для успешного использования метода в таких условиях требуется задание "хорошего" начального приближения. Существенным недостатком метода Ньютона является то, что он, хотя и сходится за меньшее число итераций, чем рассмотренные выше методы первого порядка, но требует значительно больших вычислительных затрат на каждой итерации, связанных с вычислением матрицы вторых частных производных. Поэтому метод может быть рекомендован лишь для решения тех задач, в которых вычисления матрицы Гессе производятся сравнительно легко.

Методы нулевого порядка не используют производных минимизированной функции, поэтому их следует применять, когда вычисление производных в методах первого и второго порядков связано с большими трудностями.

Методы первого порядка для определения направления поиска используют градиент минимизированной функции $\nabla\Phi(P)$ - вектор, компонентами которого являются частные производные функции по оптимизируемым параметрам. Вычисление градиента представляет известные трудности, однако сходимость методов выше, чем у методов нулевого порядка.

Применение методов второго порядка весьма ограничено из-за трудностей вычисления вторых производных. Их рекомендуется использовать в случаях, когда вычисление матрицы Гессе сравнительно легко.

Литература

Основная

1. ***Болгов В.А., Яковлев В.Б.*** Основы численных методов. - М.: МИЭТ, 2001.
2. ***Ермак В.В., Перминов В.Н., Соколов А.Г.*** Рабочие станции в проектировании БИС. - М.: Высш. шк., 1990.
3. ***Казеннов Г.Г., Соколов А.Г.*** Принципы и методология построения САПР БИС. - М.: Высш. шк., 1990.
4. ***Казеннов Г.Г.*** Основы проектирования интегральных схем и систем. - М.: БИНОМ. Лаборатория знаний, 2005.

Дополнительная

5. ***Каханер Д., Моулер К., Неш С.*** Численные методы и программное обеспечение. - М.: Мир, 2001.

6. ***Норенков И.П., Маничев В.Б.*** Основы теории и практика САПР. - М.: Высш. шк., 1990.

Приложение 1

Способы хранения разреженных матриц

Разреженные матрицы целесообразно хранить таким образом, чтобы обеспечить экономию памяти, простоту доступа к любому элементу матрицы и уменьшение числа операций, необходимых для преобразования матрицы в процессе решения линейной системы.

Преобразование разреженной матрицы (например, к верхней треугольной форме методом Гаусса) в общем случае ведет к появлению новых ненулевых элементов (ННЭ). Поэтому при хранении разреженной матрицы места для ННЭ должны быть зарезервированы заранее, либо схема должна позволять легко вводить новые элементы. Определение позиций ННЭ производится с помощью моделирования процесса преобразования матрицы, которое проводится один раз перед началом работы основной программы, так как структура разреженности не изменяется при анализе схемы (возможность появления нулей, как правило, игнорируется).

Разработано несколько способов, обеспечивающих компромисс между противоречивыми требованиями экономии памяти и времени вычислений. Выбор того или иного способа зависит от специфики задачи и особенностей ЭВМ, на которой решается задача.

Рассмотрим некоторые способы упакованного хранения разреженных матриц.

Первый способ. Все ненулевые элементы a_{ij} матрицы A размерности $m \times m$ записываются построчно в одномерном массиве A. Формируются два указательных массива: LJ и LI.

В массив LJ записываются номера столбцов ненулевых элементов a_{ij} в том же порядке, в каком элементы a_{ij} записаны в массиве A. В массив LI записываются относительные адреса первых ненулевых элементов каждой строки массива A. Длина массивов A и LJ равна общему количеству ненулевых элементов матрицы, длина массива LI равна m.

Пример 1. Для матрицы

$$\begin{bmatrix} a_{11} & a_{12} & 0 & 0 & 0 \\ a_{21} & a_{22} & 0 & a_{24} & 0 \\ 0 & 0 & a_{33} & a_{34} & 0 \\ 0 & a_{42} & a_{43} & a_{44} & a_{45} \\ 0 & 0 & 0 & a_{54} & a_{55} \end{bmatrix}$$

массивы A, LJ и LI имеют вид

$A = a_{11}a_{12}a_{21}a_{22}a_{24}a_{33}a_{34}a_{42}a_{43}a_{44}a_{45}a_{54}a_{55}$;

LI = 1 2 1 2 4 3 4 2 3 4 5 4 5;

LI = 1 3 6 8 12.

Второй способ. Использование ***однородного координатного базиса*** при формировании уравнений схемы приводит к тому, что матрица A оказывается структурно-симметричной, т.е. если $a_{ij} \neq 0$, то и $a_{ji} \neq 0$. В этом случае удобно применять следующую схему хранения.

Диагональные элементы a_{ii} записываются в массив *AD*, ненулевые элементы a_{ij} $(j > i)$ - по строкам в массив *AU*, а ненулевые элементы a_{ij} $(i > j)$ - по столбцам в массив *AL*.

Организуются два указательных массива *LJ* и *LI*. В массив *LJ* записываются номера столбцов (строк) элементов из массива *AU* (*AL*), а в массив *LI* - относительные адреса первых в строке (столбце) ненулевых элементов из массива *AU* (*AL*).

Пример 2. Для матрицы (см. пример 1)

$AD = a_{11}a_{22}a_{33}a_{44}a_{55}$;

$AU = a_{12}a_{24}a_{34}a_{45}$;

$AL = a_{21}a_{42}a_{43}a_{54}$;

LJ = 2 4 4 5;

LI = 1 2 3 4 5.

Очевидно, преимущество этого способа хранения перед первым способом заключается в том, что массив *LJ* более чем в два раза короче. Кроме того, использование структурной симметрии позволяет значительно уменьшить количество вспомогательных операций при Гауссовом исключении. Поиск элемента с заданными индексами осуществляется по тому же алгоритму, что и в первом способе, но среднее время поиска будет в два раза меньше. Наконец, при формировании матрицы узловых проводимостей нужно определять подряд позиции четырех элементов a_{pp}, a_{pq}, a_{qp} и a_{qq}. Определение позиций диагональных элементов вообще не требует поиска, а определение позиций элементов a_{pq} и a_{qp} требует только одного поиска. Поэтому среднее время формирования матрицы будет приблизительно в восемь раз меньше, чем при использовании первого способа.

Третий способ. Описанные выше способы хранения не позволяют легко вводить новые ненулевые элементы, так как это ведет к сдвигу последующих элементов массивов. От этого недостатка свободна схема упаковки, использующая связные списки.

Поставим в соответствие каждому ненулевому элементу a_{ij} два числа: номер столбца j и относительный адрес k в массиве A следующего ненулевого элемента i-й строки. Если элемент a_{ij} - последний в строке, то $k = 0$. Порядок расположения в памяти записей (a_{ij}, j, k) несуществен. Обозначим *LJ* и *KM* массивы, в которых записываются числа j и k соответственно. Тогда для матрицы

$$\begin{bmatrix} a_{11} & a_{12} & 0 & 0 & 0 \\ a_{21} & a_{22} & 0 & a_{24} & 0 \\ 0 & 0 & a_{33} & a_{34} & 0 \\ 0 & a_{42} & a_{43} & a_{44} & a_{45} \\ 0 & 0 & 0 & a_{54} & a_{55} \end{bmatrix}$$

заполнение массивов имеет следующий вид:

	A	*LJ*	*KM*	*LM*
1	a_{11}	1	2	1
2	a_{12}	2	0	3
3	a_{21}	1	4	7
4	a_{22}	2	6	5
5	a_{42}	2	11	10
6	a_{24}	4	0	
7	a_{33}	3	8	
8	a_{34}	4	0	
9	a_{44}	4	12	
10	a_{54}	4	13	
11	a_{43}	3	9	
12	a_{45}	5	0	
13	a_{55}	5	0	

Если необходимо ввести новый ненулевой элемент, например a_{35}, достаточно к массивам *A*, *LJ* и *KM* добавить запись a_{35} 5 0, а в 8-й ячейке массива *KM* заменить 0 на 14. Кроме удобства введения ННЭ, этот способ не имеет других преимуществ перед рассмотренными выше. Наоборот, применение связных списков увеличивает затраты памяти и несколько усложняет поиск элементов.

Если имеется возможность оперировать только частью ячейки памяти, то для записи чисел *j* и *k* можно использовать одну ячейку. В таком случае можно также дополнить список адресами следующих ненулевых элементов столбца (массив *LM*). Это уменьшит количество вспомогательных операций при решении системы.

При работе с разреженными матрицами с точки зрения быстродействия эффективность программы можно характеризовать величиной $T = t_0/(t_0 + t_1),$ где t_0 и t_1 - время выполнения основных и вспомогательных операций.

Экономичность с точки зрения затрат оперативной памяти можно оценивать величиной $Q = q_0/(q_0 + q_1),$ где q_0 - количество ненулевых элементов в матрице с учетом появления ННЭ; q_1 - количество ячеек, необходимых для хранения указательных массивов и команд программы решения.

Разреженность матриц можно характеризовать средним числом ненулевых элементов ρ в строках матрицы справа от главной диагонали. Таким образом, общее количество ненулевых элементов в матрице $q_0 = (2\rho + 1)m$.

Приложение 2

Меры погрешности решения

Пусть x - точное решение СЛАУ $Ax = b$, а x^* - вычисленное решение этой же системы.

Существуют две общеупотребительные меры погрешности полученного решения x^*.

1. **Вектор ошибки**

$$e = x - x^* .(П2.1)$$

2. **Вектор невязки**

$$r = b - Ax^* = A(x - x^*) . \quad (П2.2)$$

Из формулы (П2.2) следует, что ***невязка*** - это количественная мера несоответствия между правыми и левыми частями уравнений системы при подстановке в них вычисленного решения. Теория матриц говорит, что при невырожденной матрице A из равенства нулю ошибки следует равенство нулю невязки, и наоборот. Но эти два вектора не обязаны быть "малы" одновременно.

Таким образом, вычисленное решение может "почти удовлетворять" уравнениям, но совсем не походить на подлинное решение.

Напомним, что квадратная матрица A называется ***вырожденной***, если ее определитель равен нулю $\det(A) = 0$.

Если матрица A почти вырождена (т.е., если малым изменением коэффициентов уравнений систему можно сделать вырожденной), то, даже при игнорировании ошибок округления, ничтожные возмущения коэффициентов системы могут приводить к большим изменениям решения. Поэтому для таких систем уравнений нереалистично рассчитывать на то, что x^* будет хорошим приближенным решением. Если известно, что матрица далека от вырожденности, то и невязка, и ошибка будут достаточно малы.

Обусловленность матрицы

Обусловленность - это внутреннее свойство матрицы, не связанное с тем, как именно решается система уравнений. Характеризуется числом обусловленности. Число обусловленности определяется отношением:

$$\left(\mathrm{cond}A = \frac{\lambda_{\max}}{\lambda_{\min}} \right), \quad (П2.3)$$

где $\lambda_{\max}$ и $\lambda_{\min}$ - максимальное и минимальное собственные значения.

Собственные значения матрицы A можно определить как решения полиномиального уравнения $\det(A - xI = 0)$.

Проиллюстрируем физический смысл собственных значений на следующем примере.

Пример 1. Рассмотрим резистивную схему (рис.П2.1,*а*).

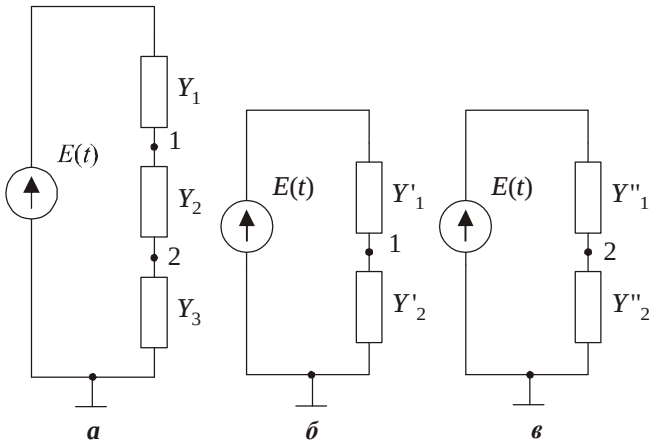


Рис.П2.1. Резистивные схемы: *а* - исходная; *б* - преобразованная для первого узла; *в* - преобразованная для второго узла

Получим ММС схемы с помощью МУП

$$\begin{vmatrix} Y_1+Y_2 & -Y_2 \\ -Y_2 & Y_2+Y_3 \end{vmatrix} \cdot \begin{vmatrix} \varphi_1 \\ \varphi_2 \end{vmatrix} = \begin{vmatrix} Y_1 E \\ 0 \end{vmatrix}.$$

Преобразуем указанную схему к виду рис.П2.1,*б*, объединив Y_2 и Y_3, так что $Y_2' = \frac{Y_2 \cdot Y_3}{Y_2 + Y}$. Тогда ММС примет вид

$$|Y_1' + Y_2'| \cdot |\varphi_1| = Y_1' E .$$

Преобразуем схему к виду рис.П2.1,*в*, аналогично объединив Y_1 и Y_2. ММС примет вид

$$|Y_2'' + Y_3''| \cdot |\varphi_2| = Y_1'' E .$$

Объединив эти уравнения, получим ММС схемы (см. рис.П2.1,*а*) в виде:

$$\begin{vmatrix} Y_1' + Y_2' & 0 \\ 0 & Y_2'' + Y_3'' \end{vmatrix} \cdot \begin{vmatrix} \varphi_1 \\ \varphi_2 \end{vmatrix} = \begin{vmatrix} Y_1' E \\ Y_1'' E \end{vmatrix}. \qquad (П2.4)$$

Здесь ненулевые элементы матрицы расположены только по главной диагонали.

Известно, что в этом случае диагональные элементы матрицы $Y_2'' + Y_3''$ и $Y_1' + Y_2'$ являются собственными значениями исходной матрицы

$$\begin{vmatrix} Y_1 + Y_2 & -Y_2 \\ -Y_2 & Y_2 + Y_3 \end{vmatrix}.$$

Легко видеть, что решение системы (П2.4) тривиально, но нетривиальна задача нахождения собственных значений.

Собственные значения для линейных резистивных схем - это собственные проводимости. При анализе переходных процессов собственные значения - это собственные постоянные времени. При частотном анализе - это собственные частоты.

Число cond(A) показывает, насколько близка к вырожденной матрица A, и насколько чувствительно решение системы $Ax = b$ к изменениям в A и b. Элементы матрицы и правой части системы линейных уравнений редко бывают известны точно. Некоторые системы возникают из эксперимента, и тогда элементы подвержены ошибкам наблюдения. Элементы других систем записываются формулами, что влечет ошибки округлений при их вычислении. Даже если систему можно точно записать в память машины, в ходе ее решения почти неизбежно будут сделаны ошибки округлений. Можно показать, что ошибки округлений в гауссовом исключении имеют то же влияние на ответ, что и ошибки в исходных элементах.

Для того чтобы уяснить смысл числа обусловленности, уточним представление о "почти вырожденности". Если A - вырожденная матрица, то для некоторых b решение x не существует, тогда как для других b оно будет неединственным. Если матрица A почти вырождена, то можно ожидать, что малые изменения в A и b вызовут очень большие изменения в x. С другой стороны, если A - единичная матрица, то b и x - один и тот же вектор. Следовательно, если матрица A близка к единичной, то малые изменения в A и b должны повлечь соответственно малые изменения в x.

Для того чтобы получить более точную и надежную меру близости к вырожденности, вводится понятие ***"норма"*** вектора. Норма - это число, которое измеряет общий уровень элементов вектора. Наиболее употребительной векторной нормой является ***евклидова длина***:

$$\left(\sum_{i=1}^{n} |x_i|^2 \right)^{\frac{1}{2}}.$$

Однако использование этой нормы сделало бы слишком трудоемкими некоторые из вычислений. Вместо нее мы определим норму вектора из n компонентов следующим образом:

$$\|x\| = \sum_{i=1}^{n} |x_i| .$$

Эта норма обладает многими из аналитических свойств евклидовой длины, а именно:

$\|x\| > 0$ $\|0\| = 0,$ если $x \neq 0$;

$\|0\| = 0$;

$\|cx\| = |c| \cdot \|x\|$ для всех скаляров c;

$\|x + y\| \leq \|x\| + \|y\|$.

Умножение вектора x на матрицу A приводит к новому вектору Ax, норма которого может очень отличаться от нормы вектора x. Это изменение нормы прямо связано с той чувствительностью, которую мы хотим определять. Область возможных изменений можно задать двумя числами:

$$M = \max_x \frac{\|Ax\|}{\|x\|};$$

$$m = \min_x \frac{\|Ax\|}{\|x\|} .$$

Максимум и минимум берутся по всем ненулевым векторам. Заметим, что если матрица A вырождена, то $m = 0$. Отношение M/m является ***числом обусловленности*** матрицы A:

$$\mathrm{cond}(A) = \frac{\max_x \frac{\|Ax\|}{\|x\|}}{\min_x \frac{\|Ax\|}{\|x\|}} .$$

Рассмотрим систему уравнений

$$Ax = b \qquad (П2.5)$$

и другую систему, полученную изменением правой части

$$A(x + \Delta x) = b + \Delta b . \qquad (П2.6)$$

Будем считать Δb ошибкой в b, а Δx - соответствующей ошибкой в x, хотя нет необходимости предполагать, что ошибки малы. Поскольку A $(\Delta x) = \Delta b$, то определения M и m немедленно ведут к неравенствам

$$\|b\| \leq M\|x\|;$$

$$\|b\| \geq m\|x\|.$$

Следовательно, при $M \neq 0$

$$\frac{\|\Delta x\|}{\|x\|} \leq \mathrm{cond}(A)\frac{\|\Delta b\|}{\|b\|}.$$

Величина $\frac{\|\Delta b\|}{\|b\|}$ есть ***относительное*** изменение правой части, а величина $\frac{\|\Delta x\|}{\|x\|}$ - ***относительная*** ошибка, вызванная этим изменением. Использование относительных изменений имеет то преимущество, что они безразмерны, т.е. нечувствительны к общим масштабирующим множителям.

Полученное неравенство показывает, что число обусловленности выполняет роль коэффициента увеличения относительной ошибки. Изменения правой части могут повлечь изменения в решении, бóльшие в cond(A) раз. То же самое справедливо в отношении изменений в элементах матрицы.

Некоторые из основных свойств числа обусловленности выводятся просто. Ясно, что $M \geq m$ и потому

$$\mathrm{cond}(A) \geq 1.$$

Если P - матрица перестановки, то компоненты вектора Px отличаются от компонентов вектора x лишь порядком. Отсюда следует, что $||Px|| = ||x||$ для всех x и, значит,

$$\mathrm{cond}(p) = 1.$$

В частности, cond(I) = 1. Если A умножается на скаляр c, то и M, и m умножаются на модуль этого скаляра, так что

$$\mathrm{cond}(cA) = \mathrm{cond}(A).$$

Если D - диагональная матрица, то

$$\mathrm{cond}(D) = \frac{\max|d_{ii}|}{\min|d_{ii}|}.$$

Так, для матрицы

$$D = \begin{pmatrix} 1 & & & & \\ & 2 & & & \\ & & 3 & & \\ & & & 4 & \\ & & & & 5 \end{pmatrix}$$

имеем $M = 5$ и $m = 1$, поэтому cond(D) = 5/1 = 5.

Последние два свойства в известной мере объясняют, почему cond(A) является лучшей мерой близости к вырожденности, чем определитель матрицы A.

В качестве примера рассмотрим диагональную матрицу порядка 100 с числом 0,1 на главной диагонали. Тогда $\det(A) = 10^{-100}$, что обычно считается малым числом. Но $\mathrm{cond}(A) = 1$, и компоненты вектора Ax отличаются от соответствующих компонентов вектора x лишь множителем 0,1. Для линейных систем уравнений такая матрица A ведет себя скорее как единичная, а не как вырожденная.

Проиллюстрируем влияние ***числа обусловленности*** на решение следующим примером.

Пример 2. Пусть для $Ax = b$

$$A = \begin{vmatrix} 4{,}1 & 2{,}8 \\ 9{,}7 & 6{,}6 \end{vmatrix}, \quad b = \begin{vmatrix} 4{,}1 \\ 9{,}7 \end{vmatrix}, \text{ тогда } \quad x = \begin{vmatrix} 1 \\ 0 \end{vmatrix}, \quad \|b\| = 13{,}8, \quad \|x\| = 1.$$

Если заменить правую часть системы на

$$b' = \begin{vmatrix} 4{,}11 \\ 9{,}70 \end{vmatrix},$$

то решением будет вектор

$$x' = \begin{vmatrix} 0{,}34 \\ 0{,}97 \end{vmatrix}.$$

Пусть $\Delta b = b - b'$ и $\Delta x = x - x'$. Тогда $\|\Delta b\| = 0{,}01$, $\|\Delta x\| = 1{,}63$.

Очень малое возмущение, внесенное нами в вектор b, совершенно изменило x. Действительно, относительные изменения равны

$$\frac{\|\Delta b\|}{\|b\|} = 0{,}0007246, \quad \frac{\|\Delta x\|}{\|x\|} = 1{,}63.$$

Поскольку $\mathrm{cond}(A)$ характеризует максимальное возможное увеличение ошибки, то

$$\mathrm{cond}(A) \geq \frac{1{,}63}{0{,}0007246} = 2249{,}4.$$

На самом деле выбранные b и Δb как раз и дают максимум, так что для этого примера $\mathrm{cond}(A) = 2249{,}4$.

Пример 3. Предположим, что мы хотим решить систему, в которой $a_{1,1} = 0{,}1$, а все остальные элементы в A и b - целые числа, и $\mathrm{cond}(A) = 10^5$. Предположим далее, что у нас компьютер с 24 битами для дробной части числа и что мы каким-то образом умеем вычислять точное решение для системы, уже записанной в память машины. Тогда единственная ошибка будет связана с двоичным представлением числа 0,1, и тем не менее можно ожидать, что $\dfrac{\|\Delta x\|}{x} \approx \mathrm{cond}(A) \cdot 2^{-24} \approx 6 \cdot 10^{-3}$.

Другими словами, простой акт записи матрицы коэффициентов в машине может вызвать изменения в третьей значащей цифре компонентов правильного решения. Мы можем подытожить сказанное следующим практическим правилом: ***при решении системы линейных уравнений относительная погрешность решения пропорциональна относительным погрешностям матрицы коэффициентов системы и ее правой части, причем константа пропорциональности равна числу обусловленности.***

Основной результат в исследовании ошибок округления в ***гауссовом исключении*** принадлежит Дж.Х. Уилкинсону. Он доказал, что вычисленное решение x^* точно удовлетворяет системе $(A + E)x^* = b$, где E - матрица, элементы которой имеют величину порядка ошибок округления в элементах матрицы A. Тем самым все ошибки округления могут быть слиты воедино и рассматриваться как единственное возмущение, внесенное в матрицу в момент ее записи в память компьютера; само исключение осуществляется без ошибок. Запись почти любой матрицы в память сопровождается возмущениями, сравнимыми с E, поэтому гауссово исключение представляет собой идеальный алгоритм решения системы $Ax = b$.