

Министерство образования и науки Российской Федерации
Томский государственный университет
систем управления и радиоэлектроники

А.О. Мелкозеров, Р.И. Аширбакиев

Компьютерное моделирование и оптимизация
электромагнитной совместимости бортовой
аппаратуры космических аппаратов

Томск
Издательство ТУСУРа
2013

УДК 629.78.054:621.391.823:004

ББК 39.66

М477

Рецензент

Спицин В.Г., д-р техн. наук, проф. Национального
исследовательского Томского политехнического университета

**Издание осуществлено в рамках реализации
Программы стратегического развития ТУСУРа
на 2012–2016 гг. и постановления Правительства РФ
от 09.04.2010 № 218, договор от 12.02.2013 № 02.G25.31.0042**

Мелкозеров А.О.

М477 Компьютерное моделирование и оптимизация электромагнитной совместимости бортовой аппаратуры космических аппаратов: моногр. / А.О. Мелкозеров, Р.И. Аширбакиев. – Томск : Изд-во Томск. гос. ун-та систем упр. и радиоэлектроники, 2013. – 220 с.

ISBN 978-5-86889-642-2

Представлена новая эволюционная стратегия $(\lambda)_{opt}$ - σ СА-ЭС, выполнен ее анализ для сферической модели целевой функции, получена формула для расчета параметра самоадаптации τ . Разработаны геометрические модели конструктивных элементов печатных плат и корпусов бортовой аппаратуры космических аппаратов в форме, пригодной для обработки алгоритмами анализа электромагнитной совместимости, а также система компьютерного моделирования и оптимизации электромагнитной совместимости бортовой аппаратуры космических аппаратов. Проведено сравнение результатов натурального и вычислительного экспериментов на примере связанных линий печатной платы. Выполнена программная реализация модуля эволюционных стратегий. Предложена методика оптимизации. Приведенные результаты могут использоваться для проектирования не только бортовой аппаратуры космических аппаратов, но и всего спектра радиоэлектронных средств.

Для разработчиков радиоэлектронной аппаратуры, студентов и аспирантов радиоэлектронных специальностей.

УДК 629.78.054:621.391.823:004

ББК 39.66

ISBN 978-5-86889-642-2

© Мелкозеров А.О.,
Аширбакиев Р.И., 2013

© Томск. гос. ун-т систем упр.
и радиоэлектроники, 2013

Список обозначений

АКМ-ЭС эволюционная стратегия с адаптацией ковариационной матрицы

БА бортовая аппаратура

КА космический аппарат

ММ математическая модель

ПО программное обеспечение

ПП печатная плата

САПР система автоматизированного проектирования

УЭМ унифицированный электронный модуль

ЭМС электромагнитная совместимость

ЭС эволюционная стратегия

\mathbf{a}, \mathbf{A} векторы и матрицы

$c_{\mu/\mu,\lambda}$ коэффициент прогресса, $c_{\mu/\mu,\lambda} = e_{\mu,\lambda}^{1,0}$

$e_{\mu,\lambda}^{\alpha,\beta}$ обобщенный коэффициент прогресса

$r^{(g)}$ расстояние между центроидом g -го поколения

$s^{(g)}$ среднее значение интенсивности мутации в поколении g

s_{st}^* стационарная интенсивность мутации

α нормированный параметр самообучения, $\alpha = \tau\sqrt{N}$

$\mu/\mu_I, \lambda$ - σ СА-ЭС эволюционная стратегия с усредняющей рекомбинацией и самоадаптацией интенсивности мутации

σ интенсивность мутации

σ СА самоадаптация интенсивности мутации

τ параметр самообучения

φ скорость прогресса эволюционной стратегии

φ_{st}^* стационарная скорость прогресса

ψ отклик механизма самоадаптации эволюционной стратегии

$\omega_{l,\lambda}$ весовые коэффициенты взвешенной эволюционной стратегии

Введение

В проектировании таких сложных технических объектов, как космические аппараты (КА), обязательен учет требований по электромагнитной совместимости (ЭМС), которые постоянно ужесточаются с ростом плотности монтажа и верхней частоты спектра полезных и помеховых сигналов в бортовой аппаратуре (БА), с уменьшением уровня полезных и ростом уровня помеховых сигналов. Натурные испытания БА КА на ЭМС и ее доработка из-за неудовлетворения требованиям ЭМС значительно удорожают и замедляют проектирование, создавая серьезные препятствия для развития отечественной космической отрасли. Эти тенденции могут привести к отставанию от мирового уровня в области технологий создания ракетно-космической техники, относящихся к перечню критических технологий Российской Федерации.

Обеспечение ЭМС особенно актуально для перспективных КА. Их особенности (унифицированные электронные модули на основе технологии «система-на-кристалле», негерметичная конструкция, повышение срока активного существования до 15 лет и более), выбранные разработчиками в силу ряда неопровержимых достоинств, еще более затрудняют как обеспечение, так и моделирование ЭМС. В частности, рост плотности и, следовательно, взаимовлияний трасс перспективных унифицированных электронных модулей (УЭМ) требует тщательного анализа целостности сигналов и питания электрических цепей. Негерметичность конструкции резко ухудшает на определенных частотных диапазонах эффективность экранирования всего КА, поэтому необходимы особые подходы к моделированию

экранирования его частей. Повышение срока активного существования КА до 15 лет обуславливает необходимость подтвержденного моделированием обеспечения значительно завышенного запаса помехозащищенности, поскольку за этот срок помеховые электромагнитные воздействия могут возрасти до столь высокого уровня, который трудно даже спрогнозировать.

Вопросам моделирования ЭМС посвящены работы, выполненные под руководством таких исследователей, как Л.Н. Кечиев, К. Пауль, Т. Хюбиг, в том числе применительно к БА – В.Ю. Кириллов, С.Ф. Чермошенцев, к КА – Дж.А. Лукаш, П. Перез и Е.Д. Пожидаев.

Исследования в области математического моделирования и оптимизации ЭМС БА КА проводятся в ТУСУРе при участии авторов с 2010 г. в рамках выполнения ОКР для ОАО «Информационные спутниковые системы им. акад. М.Ф. Решетнева» по постановлению Правительства РФ № 218. Информации по аналогичным исследованиям в интересах предприятий космической отрасли за рубежом в открытом доступе нет. Схожий проект по моделированию ЭМС воздушных судов HIRF-SE финансируется с 2008 г. 7-й рамочной программой Европейского Союза. Над данным международным проектом с бюджетом 25 млн евро работает 44 организации-партнера из 11 стран ЕС, в том числе университеты, исследовательские центры и производители авиационной техники. Конечной целью проекта HIRF-SE является переход от длительных и дорогостоящих испытаний ЭМС воздушных судов к компьютерному моделированию на ранних этапах проектирования.

Таким образом, за рубежом активно ведутся исследования с целью разработки новых подходов для качественно-

го скачка в обеспечении и моделировании ЭМС воздушных судов, а в перспективе – КА зарубежных производителей. Однако доступ к результатам этих исследований ограничен кругом европейских участников таких проектов, а возможность закупки за рубежом подобных новейших технологий исключена. Поэтому для обеспечения конкурентоспособности отечественных предприятий космической отрасли необходимо самостоятельно разрабатывать программное обеспечение для математического моделирования ЭМС БА КА.

Акцент только на задачах анализа, без автоматизированного синтеза и оптимизации, часто делает проектирование неэффективным, а также оставляет невыявленными скрытые ресурсы его ускорения. Учет этого особенно важен для оптимизации ЭМС БА КА, спецификой которой является большое количество оптимизируемых параметров и локальных экстремумов целевой функции при ресурсоемкости и разнообразии задач анализа. Даже наиболее приемлемое для такой оптимизации использование эволюционных алгоритмов серьезно затруднено с ростом сложности моделирования ЭМС перспективных КА. Поэтому необходимо совершенствование существующих алгоритмов оптимизации. Особенно актуально создание алгоритмов эволюционных стратегий (ЭС), способность которых эффективно оптимизировать зашумленные целевые функции с большим количеством параметров доказана теоретически.

Вопросам оптимизации посвящены работы, выполненные под руководством таких исследователей, как А. Оуэнс, Л.А. Растрингин, Л. Фогель, в том числе оптимизации эволюционными алгоритмами – Х.-Г. Байер, Л.И. Букатова, И. Рехенберг, Н. Хансен, Х.-П. Швифель.

Безусловно, реализация конкретных решений потребует тщательной структурно-параметрической оптимизации БА КА, которая при существующих программно-аппаратных средствах крайне затратна и предполагает наличие у разработчиков большого опыта в использовании методов оптимизации. Для снижения требований к квалификации разработчиков в этой области актуальна разработка методики оптимизации ЭМС БА КА, основанной на комплексном использовании возможностей применения программного обеспечения для математического моделирования ЭМС.

Кроме того, внедрение на отечественных предприятиях космической отрасли в качестве средств анализа ЭМС БА КА существующих программных продуктов для электродинамического моделирования проблематично по причине высокой сложности и стоимости этих продуктов. Так, стоимость лицензии на систему CST Microwave Studio составляет около 8,5 млн рублей за стандартную комплектацию без модулей оптимизации и параллельных вычислений. Для более дешевых аналогов, которые не содержат всех необходимых средств для моделирования ЭМС БА КА (одновременно схемного, квазистатического и электродинамического анализа), требуется закупка разных программных продуктов и разработка методик их совместного использования. Таким образом, целесообразно разработать собственную систему моделирования ЭМС БА КА, которая при более низкой стоимости лицензии будет иметь такие важные преимущества, как ориентирование на перспективные образцы БА КА и доступные в базовой комплектации средства оптимизации и параллельных вычислений.

Авторы благодарны проректору по научной работе ТУСУРа А.А. Шелупанову за поддержку издания книги.

1. Анализ методов моделирования и оптимизации электромагнитной совместимости

1.1. Актуальность проблемы электромагнитной совместимости

В настоящее время системы компьютерного моделирования относятся к основным инструментам разработчика радиоэлектронных устройств [1]. В частности, весь процесс разработки современной бортовой аппаратуры космических аппаратов (БА КА) ведется в специализированных системах автоматизированного проектирования. Однако постоянный рост сложности и плотности компоновки БА КА приводит к увеличению требований к квалификации разработчиков, росту затрат на проектирование и тестирование систем управления КА, замедлению процесса создания перспективных КА. Эти тенденции могут привести к отставанию отечественных разработчиков КА от мирового уровня в области технологий создания ракетно-космической техники нового поколения, которые относятся к перечню критических технологий Российской Федерации. Поэтому проблема создания технологий проектирования КА нового поколения требует разработки новых комплексов программ для моделирования и оптимизации БА КА с целью улучшения их технических характеристик.

БА КА, включая низкочастотные блоки, подвержена внешним воздействиям в очень широком спектре частот. Результаты измерений, полученные со спутника на расстоянии 360 км от поверхности Земли, показывают, что напряженность электрического поля превышает 100 В/м

в области 10 ГГц, а спектр излучений простирается до 100 ГГц [2]. Следовательно, низкочастотные блоки КА находятся под воздействием высокочастотных наводок [3] и с целью обеспечения ЭМС должны быть исследованы в диапазонах частот до 1, 18, 40 и 100 ГГц [4, 5] либо с помощью натуральных экспериментов [6], либо посредством математического моделирования.

При этом уже в настоящее время компанией FID GmbH производятся генераторы с пиковой мощностью 20 ГВт (100 синхронизированных источников, входная мощность 20 кВт, вес 150 кг), которые способны создавать на расстоянии 1 км поле с напряженностью 7–10 кВ/м [7]. Ведется разработка генераторов с пиковой мощностью 200 ГВт (входная мощность 200 кВт, вес 2 т), которые способны создавать на расстоянии 1 км поле с напряженностью 50–70 кВ/м. Следовательно, в ближайшие 15 лет не исключена возможность создания генераторов, излучающих мощные электрические поля на значительно большие расстояния, что обуславливает необходимость обеспечения подтвержденного моделированием значительно завышенного запаса помехозащищенности БА перспективных КА.

БА КА состоит из унифицированных электронных модулей (УЭМ), которые должны удовлетворять требованиям механической, тепловой и радиационной стойкости, а также электромагнитной совместимости (ЭМС) [8]. В то время как подходы к обеспечению механической, тепловой и радиационной стойкости УЭМ в достаточной мере разработаны и внедрены на предприятиях-производителях КА Российской Федерации, проблемы обеспечения ЭМС УЭМ являются сравнительно новыми и стали особенно актуальны в последние годы в связи с ростом верхней частоты спектра

используемых в КА сигналов, а также внешних электромагнитных воздействий. Аналогичные задачи обеспечения ЭМС стоят перед разработчиками любой критичной аппаратуры [9], для которой существует опасность внешних электромагнитных воздействий, в том числе электромагнитного терроризма [10].

Достоверное и эффективное математическое моделирование ЭМС БА КА подразумевает использование нескольких видов анализа, в наилучшем случае трех: схемного, квазистатического и электродинамического. В первой части раздела приведен обзор существующего программного обеспечения для электродинамического моделирования, включая программные продукты, основанные на методе моментов, методе конечных разностей во временной области, методе матрицы линий передачи, методе конечных элементов, методе конечного интегрирования.

Проектирование БА КА с привлечением только средств математического моделирования ЭМС может быть неэффективно без решения задачи автоматизированной оптимизации ЭМС БА КА, в общем случае представляемой в виде целевой функции, для которой нужно найти удовлетворяющий заданным ограничениям оптимальный набор параметров. Часто эти ограничения противоречат друг другу и ведут в итоге к появлению сложных задач оптимизации с сотнями переменных. Лежащие в их основе целевые функции не имеют аналитического представления и состоят из вызовов расчетных программ, которые возвращают решения с погрешностями из-за использования аппроксимаций в вычислительных моделях.

Теоретические исследования показали, что эволюционные стратегии (ЭС), которые являются подклассом

основанных на принципах эволюции в природе методов прямого поиска и оптимизации, способны эффективно оптимизировать такие многомерные зашумленные целевые функции [11, 12]. При этом в сравнении алгоритмов оптимизации [13], проведенном в рамках семинаров «Black-Box Optimization Benchmarking» на международных конференциях по генетическим и эволюционным вычислениям 2009 и 2010 гг., ЭС показали для ряда сложных целевых функций лучшие результаты [14].

Помимо ЭС, к эволюционным алгоритмам относятся генетические алгоритмы, эволюционное программирование и другие методы, которые также используют при работе только значения целевой функции и не требуют никакой дополнительной информации (как то производные или градиенты), что делает их пригодными для оптимизации сложных целевых функций, в том числе для решения задач оптимизации ЭМС [15].

Одно из преимуществ ЭС и ГА – работа только со значениями целевой функции – является одновременно и их недостатком: если целевая функция сравнительно проста (небольшое количество параметров, во всех точках области поиска возможно вычислить градиент, низкий уровень шума), градиентные методы показывают на порядки более высокую скорость сходимости. Для работы с такими целевыми функциями эффективен алгоритм неявного фильтрации (НФ) [16], в своей основе являющийся алгоритмом наискорейшего спуска с разностными градиентами, что обуславливает его быструю сходимость, например для квадратичных целевых функций. Преимуществом НФ перед другими градиентными методами является способность работать с зашумленными целевыми функциями [17] за счет

невной фильтрации¹ высокочастотных шумовых составляющих в значениях целевой функции. Данная особенность обосновывает выбор НФ для реализации в инструментарии оптимизации ЭМС БА КА.

1.2. Методы и системы электродинамического моделирования

В программных системах для электродинамического моделирования используются различные математические методы. Наиболее широко распространены метод моментов, метод конечных разностей, метод матрицы линий передачи, метод конечных элементов и метод конечного интегрирования.

Метод моментов (method of moments, МОМ) основан на решении интегральных уравнений Максвелла и наиболее разработан в частотной области. Преимущество метода заключается в том, что дискретизируется только поверхность анализируемой структуры, но не пространство вокруг нее. Недостаток – в процессе решения формируется плотная матрица СЛАУ, что ведет к высоким затратам на ее решение.

Метод конечных разностей (finite-difference time-domain method, FDTD) работает путем дискретизации уравнений Максвелла во временной области. Достоинство метода – возможность вычислить отклик структуры в диапазоне частот за один проход алгоритма. Недостаток – величина временного шага должна быть мала по сравнению с размером структуры.

¹ В алгоритме НФ фильтрация шума в явном виде не выполняется (специальные операции или фильтры отсутствуют), а является побочным эффектом процедуры аппроксимации градиента. Это свойство и послужило основанием для выбора названия алгоритма.

В методе матрицы линий передачи (transmission line matrix method, TLM) пространство представляется в виде матрицы из соединенных между собой линий передач, каждая из которых частично отражает и частично передает импульс. Достоинства и недостатки метода аналогичны методу конечных разностей.

Метод конечных элементов (finite element method, FEM) – численный метод для нахождения приближенных решений уравнений с частными производными, в котором пространство разбивается на конечное число областей. Преимущество метода состоит в сравнительной простоте обработки сложных структур, а также в получении в процессе решения разреженной матрицы. Недостаток – чувствительность к ошибкам разбиения структуры на конечные элементы (неправильный выбор типа элементов, неадекватное представление геометрии и т.п.).

В методе конечного интегрирования (finite integration technique, FIT) пространство делится на трехмерные ячейки, после чего решаются уравнения Максвелла в интегральной форме. К достоинствам метода относится возможность моделирования произвольных характеристик материалов (дисперсия, анизотропность, нелинейность).

В табл. 1.1 приведены данные об известных системах электродинамического моделирования, сгруппированные по используемому в них методу. Рассмотрим эти системы подробнее.

Программа ЭДЭМ (Электродинамика элементов из металла) [18] позволяет исследовать электродинамические свойства структур, допускающих аппроксимацию набором проводящих поверхностей.

Таблица 1.1

Программные продукты для электродинамического моделирования

Название	Производитель	Страна
Метод моментов		
CONCEPT	Univ. Hamburg-Harburg	Германия
FEKO	EMSS	ЮАР
EMC Studio	EmCoS	Грузия
WIPL-D	WIPL-D	Сербия
SINGULA	Integr. Eng. Software Inc.	Канада
IE3D	Zeland Software	США
Метод конечных разностей		
QUICKWAVE	QWED	Польша
CONCERTO	Vector Fields	Англия
SEMCAD-X	SPEAG	Швейцария
EMPIRE	IMST GmbH	Германия
XFDTD	REMCOM	США
GEMS	Comp. and Comm. Unlim.	США
FIDELITY	Zeland Software	США
Метод матрицы линий передачи		
CST MICROSTR.	CST AG (Flomerics Gr.)	Германия
MEFiSTo	FAUSTUS Scientific Corp.	Канада
Метод конечных элементов		
HFSS	Ansoft Corp.	США
Метод конечного интегрирования		
Microwave Studio	CST AG	Германия
Другие методы		
ЭДЭМ	edem3d.ru	Россия
KARAT	ИОФ РАН	Россия

Это металлические тела, рефлекторы, экраны, замкнутые и незамкнутые оболочки произвольной формы,

а также системы из них с характерными размерами от долей до нескольких длин волн. Для аппроксимации таких структур можно использовать плоские треугольники и четырехугольники, спирали, диски, кольца и их сектора, замкнутые и незамкнутые поверхности вращения и цилиндрические поверхности, образованные кривыми второго порядка и двумерными сплайн-линиями, а также поверхности, «натянутые» на трехмерные сплайн-линии. Анализируемые системы могут содержать проводящие или импедансные пленки, а также сосредоточенные емкости и индуктивности. Исследуемые структуры могут также образовывать бесконечные периодические решетки. ЭДЭМ позволяет находить электрические и магнитные поля, возникающие в окрестности таких структур – внутри, снаружи и на любом удалении, вычислять пространственные распределения компонентов полей и плотности потока мощности, рассчитывать диаграммы направленности, находить полную излучаемую системой мощность, коэффициент направленного действия, определять фазовые центры, поляризационные характеристики и др. В основе программы лежат алгоритмы решения интегральных уравнений (ИУ), которые были созданы в результате многолетнего развития методики, предложенной авторами программы для анализа такого типа проблем. Возможности данной системы не изучались из-за ограничений, имеющих в демонстрационных версиях, и специфики программы, ориентированной только на металлические объекты. Также стоит отметить сложность использования программы и неудобный графический интерфейс.

Система КАРАТ [19] – это программа, основанная на методе «частица в ячейке» (PIC), которая предназначена для решения нестационарных электродинамических задач, име-

ющих сложную геометрию и включающих динамические и релятивистские электроны и ионы. КАРАТ хорошо подходит для моделирования сильноточных электронных приборов, таких как виркаторы, лазеры свободных электронов, гиротроны и др., а также физических явлений в лабораторной и пространственной плазме. Возможности данной системы не изучались из-за ограничений, имеющихся в демонстрационных версиях, и специфики программы.

Метод моментов

Система CONCEPT-II. Система CONCEPT-II [20] разработана в Гамбургском институте электромагнитной теории. Этот программный продукт предназначен для численного решения задач электромагнитного излучения и рассеивания в частотной области. Ядро вычислений основано на методе моментов и интегральных уравнениях для электрических (EFIE) и магнитных (MFIE) полей. CONCEPT-II используется для вычисления токов, напряжений и полей, которые являются результатом какого-либо возбуждения металлических структур или диэлектрических тел. Электромагнитные поля могут быть вычислены как в ближней, так и в дальней зонах. Данной системой исследовался, например, прямоугольный корпус с двумя щелями, возбужденный несимметричными вибраторами, и сравнивались различные результаты (S- и Z-параметры и значения электрического поля в дальней зоне), полученные путем компьютерного моделирования и измерений. Для большинства случаев получено соответствие результатов измерения и моделирования.

Система FEKO. Система трехмерного электродинамического моделирования FEKO компании EMSS (ЮАР) предназначена для анализа антенных систем и неоднородных диэлектрических сред [21]. Метод моментов в совокупности с методом физической оптики открывает широкие возможности расчета антенных систем в присутствии больших металлических объектов со сложной формой. Удобный интерфейс построения геометрии и задания возбуждающих источников позволяет рассчитывать управляемые диаграммы направленности антенных фазовых решеток.

Система EMC Studio. EMC Studio – пакет прикладных программ, предназначенный для решения сложных проблем электромагнитной совместимости [22]. Пакет EMC Studio разработан компанией EMCoS для инженеров, работающих над проблемами ЭМС в больших системах, таких как автомобили, самолеты и корабли, требующих специального подхода для ее оценки. Если антенны, антенные усилители, кабельные узлы, интегрированные схемы или большие металлические рассеиватели должны быть оценены вместе, то в большинстве случаев прямое численное решение волнового уравнения не дает удовлетворительного результата и нужны комбинированные комплексные подходы, которые и реализованы в EMC Studio. Программа содержит интерфейсные решения для пре- и постпроцессинга с высокой точностью расчетов в сложных задачах. Пакет EMC Studio также может быть использован в качестве стандартного электромагнитного решателя, например при расчете антенных характеристик.

Система WIPL-D. Система WIPL-D ориентирована на решение антенных задач [23]. Она обладает достаточно сложным интерфейсом, который делает проблематичным создание сложных объектов.

Система SINGULA. Система SINGULA в основном ориентирована на расчет эффективной площади рассеяния [24].

Система IE3D. Система IE3D предназначена для решения антенных задач, в том числе планарных [25]. Именно для таких структур наилучшим образом подходит программа IE3D, представляющая собой полноволновой инструмент для моделирования, использующий метод моментов. Программа позволяет рассчитывать S-параметры, распределение токов на поверхности антенны, диаграмму направленности в дальней зоне для антенны как элементарного излучателя и для несложной антенной решетки, распределение поля вблизи поверхности антенны и т.д. Для более сложных антенных решеток программа позволяет после полного обчета одного элемента прогнозировать поведение линейной или прямоугольной системы путем введения так называемого «фактора решетки», который учитывает в некоторой степени прохождение сигнала из антенны в антенну при различном расположении элементов.

Метод конечных разностей во временной области

Система SEMCAD-X. SEMCAD-X – программа электромагнитного и теплового моделирования, разработанная компанией Schmid & Partner Engineering (SPEAG) [26]. Программное обеспечение спроектировано так, чтобы

удовлетворить ряд потребностей: оценка электромагнитного излучения, ЭМС, проектирование антенн (зависимость входного сопротивления от частоты, КПД, коэффициент усиления, диаграммы направленности), решение задач дозиметрии, медицины и оптики. SEMCAD-X включает подпрограмму SPICE, которая моделирует электронные схемы, составленные из различных аналоговых элементов. В большинстве реальных ситуаций эти схемы взаимодействуют с физическими устройствами типа полосковых линий или антенн. Есть возможность моделирования электромагнитных проблем с использованием фактического сигнала, управляющего электромагнитной системой. Можно рассмотреть напряжение или ток в схеме.

Система EMPIRE. Empire – универсальная программа трехмерного электромагнитного моделирования полей, которая применяется для моделирования электромагнитных полей корпусов, соединительных проводов, радиаторов, волноводов, антенн и решения проблем ЭМС [27]. Особенности программного пакета Empire: есть возможность 2D/3D-импорта и экспорта данных, дистанционное управление, простая настройка моделирования структуры, автоматическое создание сетки, параметризация, удобный дизайн. Недостатком системы EMPIRE является сложность моделирования при возбуждении плоской волной.

Система XFDTD. XFDTD – система полного трехмерного электромагнитного моделирования компании Remcom, обладает интуитивным графическим пользовательским интерфейсом подготовки расчетов и представления результа-

тов [28]. Алгоритм FDTD (Yee-ячейка) решает уравнения Максвелла для электрического и магнитного полей. Алгебраические уравнения записываются во временной форме, решаются в каждой временной точке анализа и дают широкополосную частотную характеристику. Моделируемые объекты подготавливаются в собственном трехмерном графическом редакторе или импортируются из машиностроительных систем проектирования Pro/Engineer, Inventor и др.

Система GEMS. GEMS – это пакет для моделирования задач ЭМС с удобным и интуитивно понятным 3D-интерфейсом для создания модели исследуемой структуры [29]. В GEMS есть возможность параллельных вычислений на многопроцессорных компьютерах, суперкомпьютерах и кластерах. Система создана для решения большого круга задач. Это, например, моделирование массивов антенн, сложных антенн, корпусов технических объектов, микроволновых контуров и устройств, а также моделирование воздействия электромагнитных волн на живые организмы. По сравнению с другими программными продуктами, GEMS эффективно использует аппаратные возможности вычислительной системы.

Система FIDELITY. Система FIDELITY ориентирована на моделирование трехмерных диэлектрических структур, волноводных структур и структур с сильно неравномерным распределением напряженности электрического поля [30]. Система позволяет интерактивно представлять напряжения и токи во времени и производить расчеты полей в частотной области с выводом графиков. Предусмотрено

автоматическое обнаружение сходимости временных сигналов, так что разработчикам не нужно угадывать, сколько шагов необходимо задать для выполнения моделирования. Ускорение сходимости временного сигнала уменьшает время моделирования за счет увеличения шага по времени. В программе отсутствует графический интерфейс: задание геометрии происходит через обычное диалоговое окно, где осуществляется выбор одного стандартного объекта из перечня (например, заземленной пластины), а затем задаются его параметры. Графика используется только для просмотра готовой модели и визуализации сетки дискретизации.

Метод матрицы линий передачи

Система CST MICROSTRIPES. CST MICROSTRIPES – система электромагнитного моделирования, позволяющая анализировать высокочастотные электродинамические задачи [31]. С ее помощью можно проектировать и анализировать электромагнитные устройства и структуры, а также решать задачи ЭМС. Задание геометрии происходит во встроенном редакторе (на ядре ACIS) с помощью примитивов или путем импорта геометрии из другой САПР. После завершения вычислений отображаются компоненты электрического и магнитного полей, поверхностные токи и др.

Метод конечных элементов

Система HFSS. HFSS – это пакет программ, который позволяет вычислять многомодовые S-параметры и электромагнитные поля в трехмерных пассивных структурах произвольной формы [32]. Для решения уравнений электро-

динамики пакет HFSS использует метод конечных элементов. Найденные из уравнений Максвелла решения для электромагнитного поля позволяют определить характеристики сверхвысокочастотных устройств. При этом учитываются потери на излучение и потери в материалах.

Посредством пакета HFSS можно моделировать антенны, делители мощности, волноводные элементы, фильтры и трехмерные неоднородности. Ввод данных состоит из этапов создания структуры, задания параметров материалов и выбора требуемых характеристик. В результате расчета могут быть получены поля внутри и вне структур, а также S-параметры. Рассчитанные S-параметры используются для анализа линейных и нелинейных схем, в том числе с помощью других программных продуктов.

Для снижения времени моделирования в пакете HFSS применяется адаптивный метод уплотнения сетки. В рамках метода тетраэдральные ячейки создаются на основании структуры из базовых примитивов, которые входят в библиотеку пакета HFSS. Разбиение на ячейки позволяет приближенно вычислить поле для выделения областей с высокой напряженностью или с большими градиентами. Затем происходит уплотнение разбиения на ячейки в выделенных таким образом областях. Пользователь также имеет возможность выполнить адаптацию вручную с помощью инструментов графического интерфейса пакета HFSS.

Метод конечного интегрирования

Система CST Microwave Studio. CST Microwave Studio (CST MWS) предназначена для численного моделирования высокочастотных устройств (антенн, фильтров,

ответвителей мощности, планарных, многослойных структур), анализа проблем целостности сигналов и ЭМС во временной и частотных областях с использованием прямоугольной или тетраэдральной сетки разбиения соответственно [33].

В системе CST MWS реализована технология аппроксимации для идеальных граничных условий (Perfect Boundary Approximation, PBA), которая позволяет при дискретизации поверхностей использовать преимущества прямоугольных и тетраэдральных сеток. За счет технологии PBA обеспечивается рост производительности и сокращение требований к объему памяти оперативного запоминающего устройства при незначительном снижении точности вычислений.

Кроме того, в CST MWS реализован метод тонких стенок (Thin Sheet Technique, TST), позволяющий оптимально представить две диэлектрические части кубической ячейки, разделенные тонкой металлической стенкой, а также метод подсеток (Multilevel Subgridding Scheme, MSS). Благодаря последнему линии разбиения могут начинаться и заканчиваться в любой точке анализируемого объема, вследствие чего вблизи элементов произвольной формы можно получить особые конформные слои с измельченной сеткой разбиения.

Исследование возможностей систем электродинамического моделирования

С целью сравнения представленных выше систем электродинамического моделирования выполнено компьютерное моделирование тестовых структур [34].

В системе CONCEPT-II проведено компьютерное моделирование одной структуры, в системах FEKO и EMC Studio – трех структур. В качестве структур использовались: несимметричный вибратор (НВ) на идеально проводящей пластине, металлический корпус с двумя щелями в диапазоне частот от 0,2 до 1 ГГц (для сравнения с известными результатами) и корпус блока питания (БП) с большим количеством щелей в диапазоне частот от 0,2 до 1,4 ГГц, возбужденные несимметричным вибратором. В последнем случае выполнены измерения при трех разных длинах вибратора на двух установках (в разных лабораториях): «Обзор-102» и «Обзор-103». Для большинства случаев получено хорошее соответствие между измерением и моделированием модуля входного импеданса и коэффициента отражения. Следует отметить, что системы, основанные на методе моментов, предъявляют высокие требования к компьютерным ресурсам. Главным недостатком данных программ являются большие временные затраты при широкополосном моделировании, поскольку вычисления необходимо производить в каждой частотной точке.

В системах CONCERTO и XFDTD выполнено компьютерное моделирование несимметричного вибратора на идеально проводящей пластине и металлического корпуса с двумя щелями. В системах SEMCAD-X и EMPIRE, помимо упомянутых выше двух структур, выполнено моделирование корпуса блока питания, возбужденного несимметричным вибратором (так же как и в случае с EMC Studio и FEKO).

Вычисления в QUICKWAVE-3D и GEMS не производились из-за ограничений, имеющихся в демонстрационных версиях. В системе FIDELITY вычисления производились

только на примере несимметричного вибратора на идеально проводящей плоскости из-за сложности задания геометрии.

В системе HFSS выполнено моделирование несимметричного вибратора на проводящей пластине и несимметричного вибратора внутри металлического корпуса с двумя щелями в диапазоне частот от 0,2 до 1 ГГц с целью получения модуля входного импеданса вибратора и коэффициента передачи. Показано, что задание сетки дискретизации по умолчанию может приводить к некорректным результатам. Так, для структуры вибратора на пластине имеются незначительные отличия от результатов других программ, например EMC Studio. В случае блока с двумя щелями увеличение плотности сетки ближе к краям корпуса привело к практически совпадающим результатам с FEKO и значительным расхождениям с результатами, полученными для заданной по умолчанию сетки. Главным же недостатком данной программы, как и программ, основанных на методе моментов, являются большие временные затраты при широкополосном моделировании, поскольку вычисления необходимо производить в каждой частотной точке.

В системе CST Microwave Studio производилось моделирование трех структур, возбуждаемых несимметричным вибратором, результаты которого сравнивались с ранее полученными результатами. В данной системе создавались геометрические модели и проводился электродинамический анализ: четырехслойной печатной платы 154.4010-210-ИЗМ.2-2 из соединителя; гибкого печатного кабеля; печатной платы с подсоединенным кабелем; соединителя с подсоединенным кабелем. При этом учитывалась прямоугольная рельефность верхнего диэлектрического слоя кабеля, а также слой лака на печатной плате.

Проведенный анализ позволяет сделать ряд выводов.

1. Система CST MWS является самой удобной при создании геометрии.
2. Системы CST MWS и EMPIRE наиболее пригодны для широкополосного моделирования и позволили получить самые быстрые и самые точные результаты в рассмотренных примерах.
3. В системе CST MWS достаточно легко задается возбуждение плоской волной, а также она позволяет импортировать файлы из широкого ряда систем проектирования (функция приобретается за дополнительную плату).

Таким образом, из существующего программного обеспечения система CST Microwave Studio наиболее предпочтительна для моделирования ЭМС БА КА.

1.3. Эволюционные стратегии

Эволюционные стратегии (ЭС) являются подклассом основанных на принципах эволюции в природе методов прямого поиска и оптимизации, которые используют мутацию, рекомбинацию и отбор популяций особей с целью поэтапного «выведения» все более близких к оптимальному решений [35]. Каждая особь представляет собой возможные решения задачи оптимизации. ЭС принадлежат к классу эволюционных алгоритмов, которые используют вычислительные модели некоторых известных механизмов эволюции в качестве ключевых элементов своей реализации. К эволюционным алгоритмам относятся также генетические алгоритмы, эволюционное программирование и другие методы.

Алгоритмы прямого поиска и оптимизации используют при работе только значения целевой функции и не требуют никакой дополнительной информации (как то производные или градиенты), что делает их пригодными для оптимизации сложных целевых функций при решении задач синтеза и оптимизации радиоэлектронных устройств для обеспечения их ЭМС. Зачастую такие целевые функции обладают сложным рельефом, не имеют аналитического представления, для них не выполняются условия непрерывности и выпуклости, что затрудняет применение классических методов оптимизации. ЭС способны эффективно работать с подобными целевыми функциями. По сравнению с прочими эволюционными алгоритмами, в ЭС при создании потомков основное значение имеет оператор мутации, параметры которого меняются во время оптимизации. Скорость их изменения, а также другие параметры влияют на эффективность работы ЭС со сложными целевыми функциями.

История эволюционных стратегий

История ЭС началась в 1964 г., когда с целью минимизации аэродинамического сопротивления гибкой пластины Биенерт, Рехенберг и Швевель создали первую версию ЭС [36], которая работала с одной родительской особью и одним потомком и была позже названа $(1 + 1)$ -ЭС. В 1965 г. состоялись первые компьютерные эксперименты с новым алгоритмом, а в 1968 г. выполнена оптимизация двухфазной форсунки [37], для которой была синтезирована оптимальная форма. Спустя несколько лет группа исследователей обнаружила физическое явление, которое объясняло оптимальность формы синтезированной форсунки.

Уже в самых первых научных работах ЭС были исследованы не только экспериментально, но и теоретически: в 1971 г. Рехенберг в своей диссертации [38] разработал теорию скорости сходимости $(1 + 1)$ -ЭС и на основе этих результатов предложил в 1973 г. [39] правило $1/5$ удач для адаптации интенсивности мутации. В 1974–75 гг. Швевель [40] сформулировал широко распространенные $(\mu + \lambda)$ -ЭС и (μ, λ) -ЭС и получил для них теоретические результаты для скорости сходимости. Для (μ, λ) -ЭС Швевелем была впервые предложена самоадаптация интенсивности мутации, подробно рассмотренная им далее, а также улучшенная версия ЭС с коррелированными мутациями [41].

После опубликования этих ранних работ две научные группы (под руководством Рехенберга в Берлине и Швевеля в Дортмунде) активно продолжали исследования ЭС. В частности, были предложены версии ЭС для работы с бинарными пространствами поиска, разработаны ЭС с коррелированными мутациями и проанализированы рекомбинантные ЭС.

ЭС в России не имеют такой долгой истории, так как публикации по ним на русский язык во времена СССР не переводились. Тем не менее, подобный $(1 + 1)$ -ЭС алгоритм случайного поиска с двумя особями был предложен Л.А. Растригиным в 1965 г. в статье [42], которая переведена на английский язык и известна за рубежом. В 1969 г. переводится на русский язык книга Фогеля, Оуэнса и Уолша «Искусственный интеллект и эволюционное моделирование» [43], которая познакомила массового читателя с основанными на принципах эволюции алгоритмами. В 1971 г. А.Г. Ивахненко для решения задачи синтеза перцептрона использовал метод группового учета аргумента [44], в котором

применялась селекция наиболее эффективных решений. Развитием этого метода стала схема, состоящая из трех шагов: создание потомков, вычисление целевой функции для них и отбор. Построенные по такой схеме стратегии поиска были исследованы Л.И. Букатовой [45] и широко применялись.

В 1985 г. М.А. Семенов и Д.А. Теркель предложили алгоритм, похожий на ЭС с самоадаптацией [46]. Данный алгоритм работает с набором точек, при этом каждой точке сопоставлена дисперсия случайного распределения, которая используется для генерации потомков. Однако теоретические исследования ЭС в том виде, в котором они были введены Биенертом, Рехенбергом и Швэфелем, в СССР и России не проводились.

В настоящее время развитие ЭС идет по ряду направлений. В первую очередь это создание версий ЭС с адаптацией ковариационной матрицы [47–50], к которым относятся самые высокопроизводительные ЭС, показывающие высокие результаты в сравнениях различных алгоритмов оптимизации. Основной проблемой здесь являются значительные вычислительные затраты на обновление ковариационной матрицы, а также необходимость моделирования большого количества поколений для получения элементов ковариационной матрицы, в достаточной мере соответствующих оптимизируемой целевой функции.

Вторым направлением является дальнейшее теоретическое исследование самоадаптационных ЭС. Большой вклад в этой области принадлежит профессору Х.-Г. Байеру, который в опубликованной в 2001 г. книге «Теория эволюционных стратегий» [11] предложил детерминистический метод анализа производительности (μ, λ) -ЭС с самоадапта-

цией интенсивности мутации. Данный метод позволяет получить не только качественные оценки производительности ЭС на тестовых целевых функциях, но и вывести аналитические формулы для расчета таких оптимальных значений параметров, при которых производительность ЭС будет максимальна. Высказанные в работе [11] идеи активно развиваются в настоящее время учениками профессора Байера применительно к конкретным целевым функциям. В 2002 г. Д. Арнольд по материалам диссертации опубликовано теоретическое исследование ЭС для модели зашумленной целевой функции [12]. Майер-Ниберг С. в 2007 г. провела теоретический анализ самоадаптационных ЭС с усредняющей рекомбинацией для сферической модели целевой функции [51, 52]. Финк Ш. в 2011 г. выполнил теоретический анализ ЭС для модели целевой функции в виде зависящих от параметра квадратичных форм [53]. Актуальным направлением дальнейших исследований в области теоретического анализа ЭС является анализ более сложных моделей целевых функций [54–56].

В рамках теории ЭС существует общепринятая система наименований вариантов ЭС, которая основывается на общем описании алгоритма канонических ЭС.

Канонические варианты эволюционных стратегий

Схематически алгоритм ЭС можно представить в следующем виде [35].

1. Инициализировать популяцию $P_\mu = \{a_1, \dots, a_\mu\}$ из μ родительских особей.

2. Сформировать λ потомков \tilde{a} , которые образуют популяцию потомков $\tilde{P}_\lambda = \{\tilde{a}_1, \dots, \tilde{a}_\lambda\}$, где для создания каждого потомка \tilde{a} необходимо:
 - отобрать (случайным образом) ρ родителей из P_μ (если $\rho = \mu$, то выбрать все родительские особи);
 - выполнить рекомбинацию ρ выбранных родителей a с целью получения рекомбинантной особи g ;
 - выполнить мутацию набора s параметров стратегии рекомбинантной особи g ;
 - выполнить мутацию вектора параметров u рекомбинантной особи g с использованием мутированного набора параметров стратегии, который управляет статистическими характеристиками мутации вектора параметров.
3. Выбрать новую родительскую популяцию либо из популяции потомков \tilde{P}_λ (отбор *comma-selection*), либо из популяции потомков \tilde{P}_λ и родительской популяции P_μ (отбор *plus-selection*).
4. Перейти к шагу 2, пока условие завершения работы алгоритма не будет выполняться.

Представленный алгоритм описывает оба канонических варианта ЭС – $(\mu/\rho, \lambda)$ -ЭС с отбором *comma-selection* и $(\mu/\rho + \lambda)$ -ЭС с отбором *plus-selection*. Далее рассматриваются ЭС с отбором *comma-selection*. К данному виду ЭС относится и стандартная ЭС с усредняющей рекомбинацией и самоадаптацией интенсивности мутации.

Стандартная эволюционная стратегия с усредняющей рекомбинацией и самоадаптацией интенсивности мутации

Стандартная ЭС с усредняющей рекомбинацией и самоадаптацией интенсивности мутации (сокращенно $(\mu/\mu_I, \lambda)$ - σ СА-ЭС) использует при создании потомка все родительские особи. Нижний индекс I означает промежуточную рекомбинацию, при которой рекомбинантная особь рассчитывается как центроид родительской популяции, то есть параметры стратегии и вектор параметров рекомбинантной особи равны среднему значению этих переменных всех родительских особей.

В виде псевдокода алгоритм $(\mu/\mu_I, \lambda)$ - σ СА-ЭС (алгоритм 1.1) можно представить следующим образом.

В строках 1–3 алгоритма 1.1 задается начальная интенсивность мутации $\sigma^{(0)}$, родительский вектор параметров $y^{(0)}$ и счетчик поколений g . После этого в строках 5–10 создается λ потомков следующим образом: сначала в строке 6 генерируется интенсивность мутации $\tilde{\sigma}_l$ l -го потомка путем мутации $\sigma^{(g)}$ с помощью логарифмически нормального оператора $\exp[\tau \mathcal{N}_l(0, 1)]$, где значение параметра самообучения τ определяет скорость самоадаптации $\sigma^{(g)}$.

Поскольку во внешних механизмах для адаптации интенсивности мутации необходимости нет, этот вариант получил название σ -самоадаптации (σ СА). В строке 7 создается $\mathcal{N}(0, \mathbf{I})$ -нормально распределенный случайный вектор z_l , который затем используется для вычисления вектора параметров \tilde{y}_l в строке 8. В строке 9 вычисляется значение целевой функции \tilde{F}_l в точке \tilde{y}_l .

Алгоритм 1.1 Алгоритм $(\mu/\mu_I, \lambda)$ -σСА-ЭС

```
1  $\sigma^{(0)} \leftarrow \sigma_{init}$ 
2  $y^{(0)} \leftarrow y_{init}$ 
3  $g = 1$ 
4 выполнять
5   для  $l = 1, \dots, \lambda$  начать
6      $\tilde{\sigma}_l \leftarrow \sigma^{(g)} e^{\tau \mathcal{N}_l(0,1)}$ 
7      $\tilde{z}_l \leftarrow \mathcal{N}_l(\mathbf{0}, \mathbf{I})$ 
8      $\tilde{y}_l \leftarrow y^{(g)} + \tilde{\sigma}_l \tilde{z}_l$ 
9      $\tilde{F}_l \leftarrow F(\tilde{y}_l)$ 
10  конец
11   $\tilde{F}_{sort} \leftarrow \text{sort}(\tilde{F}_{1\dots\lambda})$ 
12   $\sigma^{(g+1)} \leftarrow \frac{1}{\mu} \sum_{m=1}^{\mu} \tilde{\sigma}_{m;\lambda}$ 
13   $\langle y \rangle \leftarrow \frac{1}{\mu} \sum_{m=1}^{\mu} \tilde{y}_{m;\lambda}$ 
14   $y^{(g+1)} \leftarrow \langle y \rangle$ 
15   $g = g + 1$ 
16 пока условие завершения работы алгоритма не выполняется
```

В строке 11 выполняется сортировка \tilde{F}_l потомков, в строке 12 – расчет арифметического среднего $\tilde{\sigma}_l$ лучших μ потомков. Затем в строках 13–14 вычисляется новый родительский вектор параметров $y^{(g+1)}$. После этого инкрементируется счетчик поколений и проверяется условие завершения работы алгоритма. Если условие не выполняется, происходит возврат на строку 4, иначе алгоритм завершается. После выполнения условия завершения работы алгоритма текущая точка поиска рассматривается как приближенное значение оптимума целевой функции $F(y)$.

Для теоретического анализа ЭС предложен метод динамических систем [11]. По сравнению с другими методами теоретического анализа ЭС, преимущество данного метода заключается в том, что полученные с его помощью формулы для описания поведения ЭС содержат все множители в явном виде и позволяют более точно оценивать производительность ЭС.

Анализ эволюционных стратегий

При анализе ЭС с использованием метода динамических систем алгоритм ЭС рассматривается с точки зрения математической теории динамических систем [57]. Цель анализа – предсказать поведение (динамику) ЭС в долгосрочной перспективе при условии, что начальное состояние ЭС задано и описываемый динамическими уравнениями ЭС закон эволюции по времени не изменяется в процессе выполнения оптимизации.

Входные данные для анализа включают начальную интенсивность мутации $\sigma^{(0)}$ (для σ СА-ЭС, параметры стратегии $\mathbf{s}^{(0)}$ в общем случае), вектор параметров $\mathbf{y}^{(0)} = (y_1^{(0)}, y_2^{(0)}, \dots, y_N^{(0)})^T$ и целевую функцию $F(\mathbf{y})$. С помощью этой информации задаются динамические уравнения в конечных разностях, которые описывают ожидаемое изменение состояния ЭС между поколениями g и $g + 1$:

$$\begin{cases} \sigma^{(g+1)} = \sigma^{(g)} (1 + \psi(\sigma^{(g)}, \mathbf{y}^{(g)})) + \varepsilon_\sigma(\sigma^{(g)}, \mathbf{y}^{(g)}); \\ y_1^{(g+1)} = y_1^{(g)} - \varphi_1(\mathbf{s}^{(g)}, \mathbf{y}^{(g)}) + \varepsilon_{\varphi_1}(\mathbf{s}^{(g)}, \mathbf{y}^{(g)}); \\ y_2^{(g+1)} = y_2^{(g)} - \varphi_2(\mathbf{s}^{(g)}, \mathbf{y}^{(g)}) + \varepsilon_{\varphi_2}(\mathbf{s}^{(g)}, \mathbf{y}^{(g)}); \\ \vdots \\ y_N^{(g+1)} = y_N^{(g)} - \varphi_N(\mathbf{s}^{(g)}, \mathbf{y}^{(g)}) + \varepsilon_{\varphi_N}(\mathbf{s}^{(g)}, \mathbf{y}^{(g)}), \end{cases} \quad (1.1)$$

где ψ и φ_i – ожидаемые изменения соответственно интенсивности мутации и i -х компонентов вектора параметров; ε_σ и ε_{φ_i} – соответствующие стохастические флуктуационные составляющие.

Предполагается, что данные уравнения позволяют получить практически значимые формулы для расчета оптимальных параметров ЭС. Асимптотически точные формулы для ψ и φ_i можно получить при допущении, что размерность пространства поиска стремится к бесконечности и флуктуационные составляющие ε_σ и ε_{φ_i} не учитываются. Заметим, что вывод точных формул для ψ и φ_i в соответствии с текущим уровнем развития теоретических исследований ЭС невозможен, так как лежащие в их основе уравнения Колмогорова–Чепмена, формально описанные в [11], до настоящего времени не решены.

Помимо $(\mu/\mu_I, \lambda)$ - σ СА-ЭС и $(\lambda)_{\text{opt}}$ - σ СА-ЭС, в приложении к задачам оптимизации ЭМС целесообразно рассмотреть ЭС с адаптацией ковариационной матрицы, показывающие высокие результаты в сравнениях алгоритмов оптимизации.

Эволюционные стратегии с адаптацией ковариационной матрицы

В последние годы ЭС с адаптацией ковариационной матрицы [58] привлекли большое внимание исследователей благодаря высокой производительности, которую продемонстрировали как в синтетических сравнениях [14], так и в реальных задачах [59]. Рассмотрим ЭС с адаптацией ковариационной матрицы (АКМ-ЭС, covariance matrix adaptation evolution strategy, CMA-ES)

и σ -самоадаптивной АКМ-ЭС (АКМ- σ СА-ЭС, covariance matrix self-adaptation evolution strategy, CMSA-ES).

$(\mu/\mu_W, \lambda)$ -АКМ-ЭС. $(\mu/\mu_W, \lambda)$ -ЭС с адаптацией ковариационной матрицы использует взвешенную рекомбинацию, которая обозначается индексом W в скобках перед названием стратегии. Алгоритм $(\mu/\mu_W, \lambda)$ -АКМ-ЭС представлен ниже [60].

Алгоритм 1.2 Алгоритм $(\mu/\mu_W, \lambda)$ -АКМ-ЭС

```

1  $\sigma \leftarrow \sigma_{init}$ 
2  $y \leftarrow y_{init}$ 
3 выполнять
4   для  $l = 1, \dots, \lambda$  начать
5      $s_l \leftarrow \sqrt{C} \mathcal{N}_l(\mathbf{0}, \mathbf{I})$ 
6      $y_l \leftarrow y + \sigma s_l$ 
7      $F_l \leftarrow F(y_l)$ 
8   конец
9    $y \leftarrow y + \sigma \langle \mathbf{s} \rangle_W$ 
10   $\mathbf{p} \leftarrow \left(1 - \frac{1}{\tau_p}\right) \mathbf{p} + \sqrt{\frac{1}{\tau_p} \left(2 - \frac{1}{\tau_p}\right)} \sqrt{\mu_{eff}} \langle \mathbf{s} \rangle_W$ 
11   $\mathbf{C} \leftarrow \left(1 - \frac{1}{\tau_p}\right) \mathbf{C} + \frac{1}{\tau_p} \left[ \frac{1}{\mu_{eff}} \mathbf{p} \mathbf{p}^T + \left(1 - \frac{1}{\mu_{eff}}\right) \langle \mathbf{s} \mathbf{s}^T \rangle_W \right]$ 
12   $\mathbf{p}_\sigma \leftarrow \left(1 - \frac{1}{\tau_p}\right) \mathbf{p}_\sigma + \sqrt{\frac{1}{\tau_p} \left(2 - \frac{1}{\tau_p}\right)} \sqrt{\mu_{eff}} \langle \mathcal{N}(\mathbf{0}, \mathbf{I}) \rangle_W$ 
13   $\sigma \leftarrow \sigma \exp \left[ \frac{\|\mathbf{p}_\sigma\| - \bar{\chi} N}{d \chi N} \right]$ 
14 пока условие завершения работы алгоритма не выполняется

```

Коррелированные мутации генерируются состоящей из двух шагов процедурой: сначала в строке 5 вектор $\mathcal{N}_l(\mathbf{0}, \mathbf{I})$, компоненты которого являются независимыми идентично распределенными стандартными нормальными

случайными величинами, умножается на матрицу \sqrt{C} . Получаемые в результате случайные векторы $z = \sqrt{C}\mathcal{N}_l(0, I)$ $\mathcal{N}_l(0, C)$ нормально распределены. Матрицу \sqrt{C} можно интерпретировать как «квадратный корень» ковариационной матрицы C . Стандартный способ получения \sqrt{C} в АКМ-ЭС основан на разложении по собственным значениям решением задачи о собственных значениях.

После создания коррелированного гауссовского вектора s его длина масштабируется в строке 6. Таким образом получается мутация bs , которая затем прибавляется к текущей точке поиска для получения нового потомка. Значение целевой функции для потомка вычисляется в строке 7. Новая текущая точка поиска вычисляется в строке 9 посредством рекомбинации μ лучших потомков, которая реализована в виде взвешенного усреднения.

Адаптация матрицы C осуществляется в строке 11 с использованием аккумулирующего вектора p и матричной оценки $\langle ss^T \rangle_W$. Строка 6 выполняет экспоненциальное сглаживание (усреднение), при котором «память» матрицы C затухает пропорционально $\left(1 - \frac{1}{\tau_p}\right)^g$, где g – счетчик поколений. В строке 13 происходит обновление текущей величины мутации.

$(\mu/\mu_I, \lambda)$ -АКМ- σ СА-ЭС. Байер и Сендхофф [61] предложили модифицированную версию АКМ-ЭС – σ -самоадаптационную АКМ-ЭС, которая сочетает простоту самоадаптационных ЭС со способностью выполнять адаптацию ковариационной матрицы.

Алгоритм 1.3 Алгоритм $(\mu/\mu_I, \lambda)$ -АКМ- σ СА-ЭС

```
1  $\sigma \leftarrow \sigma_{init}, \mathbf{y} \leftarrow \mathbf{y}_{init}$ 
2 выполнять
3   для  $l = 1, \dots, \lambda$  начать
4      $\sigma_l \leftarrow \langle \sigma \rangle e^{\tau \mathcal{N}_l(0,1)}$ 
5      $\mathbf{s}_l \leftarrow \sqrt{\mathbf{C}} \mathcal{N}_l(\mathbf{0}, \mathbf{I})$ 
6      $\mathbf{z}_l \leftarrow \sigma_l \mathbf{s}_l$ 
7      $\mathbf{y}_l \leftarrow \mathbf{y} + \mathbf{z}_l$ 
8      $F_l \leftarrow F(\mathbf{y}_l)$ 
9   конец
10   $\mathbf{y} \leftarrow \mathbf{y} + \langle \mathbf{z} \rangle$ 
11   $\mathbf{C} \leftarrow \left(1 - \frac{1}{\tau_c}\right) \mathbf{C} + \frac{1}{\tau_c} \langle \mathbf{s} \mathbf{s}^T \rangle$ 
12 пока условие завершения работы алгоритма не выполняется
```

$(\mu/\mu_I, \lambda)$ -АКМ- σ СА-ЭС с ограничениями. Практические задачи оптимизации обычно имеют ограничения и для правильной работы с ними необходимо вносить изменения либо в алгоритм ЭС, либо в целевую функцию. Значительное внимание в литературе уделено второму подходу [62, 63], тогда как тема ЭС с ограничениями рассматривается в небольшом количестве публикаций [59, 64]. Вариант σ -самоадаптивной АКМ-ЭС ($(\mu/\mu_I, \lambda)$ -АКМ- σ СА-ЭС) с ограничениями предложен в [65] для решения задачи оптимизации в области финансов.

В строке 1 алгоритма 1.4 задается начальная интенсивность мутации $\sigma^{(0)}$, родительский вектор параметров $\mathbf{y}^{(0)}$ и счетчик поколений g . После этого в строках 3–10 создается λ потомков следующим образом: сначала в строке 4 генерируется интенсивность мутации $\tilde{\sigma}_l$ l -го потомка путем мутации $\sigma^{(g)}$ с помощью логарифмически нормального

оператора $\exp[\tau \mathcal{N}_l(0, 1)]$, где значение параметра самообучения $\tau = \sqrt{2N}$ выбрано в соответствии с [61].

Алгоритм 1.4 $(\mu/\mu_I, \lambda)$ -АКМ-сСА-ЭС с ограничениями

```

1  $\sigma^{(0)} \leftarrow \sigma_{init}, \mathbf{y}^{(0)} \leftarrow \mathbf{y}_{init}, \mathbf{C} = \mathbf{I}, g = 1$ 
2 выполнить
3   для  $l = 1, \dots, \lambda$  начать
4      $\tilde{\sigma}_l \leftarrow \sigma^{(g)} e^{\tau \mathcal{N}_l(0, 1)}$ 
5      $\mathbf{z}_l \leftarrow \det(\mathbf{C})^{-\frac{1}{2(N-1)}} \sqrt{\mathbf{C}} \mathcal{N}_l(\mathbf{0}, \mathbf{I})$ 
6      $\tilde{\mathbf{s}}_l \leftarrow \text{FullfillBoxConstraints}(\mathbf{y}^{(g)}, \mathbf{M}\mathbf{z}_l, \tilde{\sigma}_l)$ 
7      $\tilde{\mathbf{y}}_l \leftarrow \mathbf{y}^{(g)} + \tilde{\sigma}_l \tilde{\mathbf{s}}_l$ 
8      $\tilde{F}_l \leftarrow F(\tilde{\mathbf{y}}_l)$ 
9      $\tilde{\mathbf{z}}_l \leftarrow \mathbf{M}^{-1} \tilde{\mathbf{s}}_l$ 
10  конец
11   $\tilde{\mathbf{F}}_{sort} \leftarrow \text{sort}(\tilde{F}_{1\dots\lambda})$ 
12   $\langle \tilde{\sigma} \rangle \leftarrow \frac{1}{\mu} \sum_{m=1}^{\mu} \tilde{\sigma}_{m;\lambda}$ 
13   $\sigma^{(g+1)} \leftarrow \begin{cases} \langle \tilde{\sigma} \rangle & \langle \tilde{\sigma} \rangle \leq 1/\sqrt{N} \\ 1/\sqrt{N} & \text{otherwise} \end{cases}$ 
14   $\mathbf{y}^{(g+1)} \leftarrow \frac{1}{\mu} \sum_{m=1}^{\mu} \tilde{\mathbf{y}}_{m;\lambda}$ 
15   $\mathbf{C} \leftarrow \left(1 - \frac{1}{\tau_c}\right) \mathbf{C} + \frac{1}{\tau_c} \langle \tilde{\mathbf{z}} \tilde{\mathbf{z}}^T \rangle$ 
16   $g = g + 1$ 
17 пока условие завершения работы алгоритма не выполняется

```

В строке 5 создается $\mathcal{N}(0, \mathbf{C})$ -нормально распределенный случайный вектор \mathbf{z}_l , который затем умножается на матрицу \mathbf{M} для обеспечения ограничения в виде равенства $\sum_{i=1}^N (\tilde{\mathbf{y}}_l)_i = 1$. Матрица \mathbf{M} состоит из $(N-1) \times (N-1)$ единичной матрицы, после последней строки которой добавлена вектор-строка с $(N-1)$ компонентами, равны-

ми -1 [65]. Полученный вектор $s_l = Mz_l$ проверяется на нарушение двухсторонних ограничений $0 \leq (\tilde{y}_l)_i \leq 1$ в строке 6. Функция `FullfillBoxConstraints` реализуется как [65]

$$\tilde{s}_l = \begin{cases} s_l & (\mathbf{y}^{(g)} + \tilde{\sigma}_l s_l)_i \geq 0 \forall i \in [1, N], \\ (\check{\mathbf{y}} - \mathbf{y}^{(g)}) / \tilde{\sigma}_l & \text{otherwise,} \end{cases} \quad (1.2)$$

где $\check{\mathbf{y}}$ – вектор параметров допустимого потомка, который находится на минимальном расстоянии от недопустимого потомка с вектором параметров $\mathbf{y}^\dagger = \mathbf{y}^{(g)} + \tilde{\sigma}_l s_l$. Недопустимый потомок \mathbf{y}^\dagger исправляется с помощью алгоритма, предложенного в [65] (алгоритм 1.5).

Алгоритм 1.5 Алгоритм для исправления недопустимого потомка \mathbf{y}^\dagger

- 1 $\mathbf{d} \leftarrow \text{SortDescending}(\mathbf{y}^\dagger)$, $\Delta \leftarrow 0$
 - 2 для $i = 1, \dots, N$ начать
 - 3 $\Delta \leftarrow \Delta + (\mathbf{d})_i$
 - 4 если $(\mathbf{d})_i - (\Delta - 1) / i > 0$ то $\gamma \leftarrow (\Delta - 1) / i$
 - 5 конец
 - 6 $\check{\mathbf{y}} \leftarrow (\max(\mathbf{y}_1^\dagger - \gamma, 0), \dots, \max(\mathbf{y}_N^\dagger - \gamma, 0))^T$
-

После проверки на двухсторонние ограничения для потомков вычисляются векторы параметров \tilde{y}_l значения целевой функции \tilde{F}_l . Так как вектор \tilde{s}_l мог быть изменен в (1.2), он трансформируется обратно в \tilde{z}_l в строке 9 алгоритма 1.4.

В строке 11 выполняется сортировка \tilde{F}_l потомков, в строке 12 – расчет арифметического среднего $\tilde{\sigma}_l$ лучших μ потомков. Среднее значение $\langle \tilde{\sigma} \rangle$ ограничивается сверху значением $1/\sqrt{N}$ в строке 13, так как большая по величине интенсивность мутации ведет к потомкам, которые лежат вне области допустимых решений [65]. Затем в строке 14

вычисляется новый родительский вектор параметров $y^{(g+1)}$ и в строке 15 обновляется ковариационная матрица C с учетом вектора \tilde{z}_l и временной постоянной $\tau_c = 1 + 0,5(N - 1)N/\mu$. После этого инкрементируется счетчик поколений и проверяется условие завершения работы алгоритма. Если условие не выполняется, происходит возврат на строку 2, иначе алгоритм завершается.

1.4. Обоснование необходимости создания собственной системы моделирования и оптимизации

Разработка системы моделирования ЭМС БА КА является комплексной проблемой проектирования программного обеспечения высокой сложности, для решения которой необходима совместная работа коллектива программистов и специалистов по ЭМС на протяжении многих лет. Однако проблемы подобного уровня решаемы, что продемонстрировала компания CST AG, которая была основана в 1992 г. сотрудниками Дармштадтского технического университета (Германия) и в настоящее время является лидером международного рынка программных продуктов для электродинамического моделирования.

По результатам анализа существующего программного обеспечения для моделирования ЭМС БА КА наиболее предпочтительна система CST MWS компании CST AG. Однако система CST MWS имеет два основных недостатка.

1. Значительная сложность использования, требующая либо понимания на уровне эксперта теории электромагнитного поля, ограничений используемых методов и моделей, либо разработки на основе экспертных знаний

подробных пошаговых методик моделирования ЭМС БА КА средствами CST MWS.

2. Высокая стоимость лицензий на продукты фирмы CST: около 8,5 миллионов рублей по прайс-листу российского дистрибьютора (см. последнюю строку табл. 1.2, где указана цена в рублях по курсу 37 руб./евро с НДС 18 %) за минимальную комплектацию без модулей для оптимизации и параллельных вычислений [10].

Таблица 1.2

Цены на программные продукты фирмы CST, евро (без НДС)

Название продукта	Версия	Сопровождение	Итого
CST MWS (T+E+F)	97297	19459	116756
DS with MWS	13900	2780	16680
EMS ES+MS	50965	10193	61158
Итого	162162	32432	194594
Итого в руб. с НДС	7079993	1415981	8495974

Учитывая, что на большом предприятии космической или оборонной отрасли программным обеспечением для моделирования ЭМС необходимо оснащать десятки рабочих мест, высокая стоимость лицензий является серьезной финансовой проблемой [66] и исключает вариант закупки таких дорогостоящих зарубежных систем, как CST MWS. Более дешевые аналоги не содержат всех необходимых средств для моделирования ЭМС БА КА (одновременно схемного, квазистатического и электродинамического анализа) в одном программном продукте, что делает необходимыми закупку нескольких программных продуктов и разработку методик их сопряжения и совместного использования.

Таким образом, разработка собственной системы моделирования ЭМС БА КА целесообразна по финансовым соображениям, однако данный вариант обладает также рядом других важных преимуществ [10].

1. Тесное взаимодействие между разработчиками и пользователями системы моделирования ЭМС БА КА, обеспечивающее эффективное сопровождение и модернизацию системы.
2. Ориентированное на перспективные образцы БА КА (унифицированные электронные модули по технологии «система-на-кристалле») развитие системы моделирования ЭМС БА КА.
3. Применение новейших теоретических моделей, программная реализация которых отсутствует в существующем коммерческом программном обеспечении.
4. Возможность опередить зарубежные программные продукты по некоторым характеристикам, в первую очередь в области методов оптимизации и параллельных вычислений.

Один из авторов данной работы в августе 2011 г. лично посетил штаб-квартиру компании CST AG в г. Дармштадте и встретился с директором по разработке этой компании, начальниками отделов разработки модулей низкочастотного моделирования и графической оболочки, а также директором по персоналу. Из знакомства с процессом разработки CST MWS можно сделать вывод, что для создания сходного по возможностям программного продукта для моделирования ЭМС с использованием аналогичных технологий необходимо выполнить работу в объеме нескольких десятков человеко-лет. При этом цена лицензии на такой продукт

будет сопоставима с ценой лицензии CST MWS. Таким образом, для разработки системы моделирования ЭМС БА КА необходимо использовать альтернативные системе CST MWS подходы к проектированию программного обеспечения.

Во-первых, все функции системы CST MWS доступны в первую очередь через оконный графический интерфейс, возможность вызова функций через язык скриптов реализована в виде надстройки и играет второстепенную роль. Такая технология позволяет сравнительно быстро создать удобный пользовательский интерфейс, однако усложняет поддержку кода (например, в коде вычислительных модулей системы CST MWS вызываются функции устаревшей библиотеки оконного интерфейса MFC, что делает задачу замены данной библиотеки на другую весьма трудоемкой), а также реализацию и отладку новых функций (необходимо одновременно реализовывать и отлаживать логику графического интерфейса). Для системы моделирования ЭМС БА КА целесообразно использовать альтернативную технологию: доступ ко всем функциям предоставить через интерпретируемый язык скриптов, а графический интерфейс реализовать в виде надстройки.

Во-вторых, чтобы исключить зависимость кода вычислительных модулей от библиотек оконного интерфейса или матричных операций и обеспечить возможность простой замены данных библиотек на аналоги, целесообразно реализовать ядро системы моделирования ЭМС БА КА, которое предоставляет стандартный объектно-ориентированный интерфейс к библиотекам оконного интерфейса и матричных операций, а также низкоуровневым функциям операционной системы.

В-третьих, для учета особенностей моделирования ЭМС БА КА необходимо реализовать набор специализированных команд языка скриптов, с помощью которых пользователь сможет создавать математические модели конструктивных элементов печатных плат и корпусов УЭМ БА КА. При этом синтаксис команд должен быть достаточно прост для генерации системой моделирования ЭМС БА КА математических моделей в автоматическом режиме при импорте из форматов САПР печатных плат.

В-четвертых, чтобы вычислительные модули поддерживали возможность параллельных вычислений в несколько потоков на многоядерных процессорах, ядро системы моделирования ЭМС БА КА должно быть реентерабельным [67].

Задача оптимизации ЭМС БА КА имеет значительную сложность, так как соответствующая целевая функция предполагает моделирование ЭМС с большим числом параметров и в широком диапазоне частот, что связано с высокими вычислительными затратами. Поэтому для получения удовлетворяющих заданным критериям результатов оптимизации за приемлемое время необходимо минимизировать количество вычислений целевой функции.

Одним из способов снижения необходимого количества вычислений целевой функции при оптимизации ЭС является создание новых, улучшенных версий ЭС. Так, известный недостаток всех $(\mu/\rho \dagger \lambda)$ -эволюционных стратегий заключается в том, что они не используют всю информацию об особях-потомках, которая рассчитывается для каждого поколения во время работы эволюционной стратегии [68]. Для компенсации этого недостатка можно использовать взвешенную мультирекомбинацию [68, 69], которая

принимает во внимание информацию о взаимном упорядочении всех потомков, а также не игнорирует самые худшие особи, как это делает $(\mu/\rho + \lambda)$ -эволюционная стратегия. Влияние каждой особи на результат рекомбинации определяется ее положением после упорядочивания, которое основано на значениях целевой функции. Таким образом, для создания улучшенной версии стандартной ЭС целесообразно внедрить в ее алгоритм взвешенную рекомбинацию.

Однако для использования полученной ЭС на практике необходимо задать оптимальное значение ее параметра самообучения τ , при котором скорость приближения текущей точки поиска к оптимуму является максимальной. Для этого целесообразно выполнить анализ ЭС с помощью метода динамических систем и получить аналитическую формулу для расчета оптимальных значений τ . В рамках метода динамических систем анализ ЭС выполняют для сферической модели целевой функции

$$F_{\text{sphere}}(\mathbf{y}) = \sum_{i=1}^N y_i^2, \quad (1.3)$$

где N – размерность пространства поиска. В первом приближении предполагается, что полученные для сферической модели целевой функции формулы расчета τ также применимы для оптимизации более сложных целевых функций с большим количеством параметров, типичных для задач ЭМС.

Чтобы применить полученные для ЭС теоретические результаты для решения практических задач, необходимо выполнить программную реализацию ЭС. При этом помимо стандартной ЭС и предложенного алгоритма ЭС со взвешенной рекомбинацией целесообразно также рассмотреть

ЭС с адаптацией ковариационной матрицы, показывающие высокие результаты в сравнениях алгоритмов оптимизации.

Кроме эволюционных алгоритмов, к которым относятся ЭС, пользователь должен иметь доступ к алгоритмам оптимизации, эффективным при работе с сравнительно простыми целевыми функциями с небольшим числом параметров. Для оптимизации таких целевых функций следует выполнить программную реализацию алгоритма неявного фильтрации.

После того как пользователю системы моделирования ЭМС БА КА посредством программной реализации будет обеспечен доступ к алгоритмам оптимизации, необходимо разработать методику оптимизации ЭМС БА КА с применением реализованных видов анализа и методов оптимизации и апробировать ее на как можно более разнообразных примерах: оптимизация параметров одиночных и многопроводных трасс печатных плат, проводных структур с сосредоточенными нагрузками, математических моделей компонентов и более сложных законченных устройств (многокаскадного модального фильтра).

В итоге необходимо решить следующие задачи.

1. Предложить для целевых функций с большим количеством параметров усовершенствованный алгоритм ЭС и выполнить его анализ.
2. Разработать систему компьютерного моделирования ЭМС БА КА и выполнить ее тестирование.
3. Предложить методику оптимизации ЭМС БА КА и провести её тестирование.

Результатом решения поставленных задач должен быть программный продукт и методика.

2. Алгоритмы и модели для моделирования и оптимизации электромагнитной совместимости бортовой аппаратуры космических аппаратов

2.1. Алгоритм эволюционной стратегии со взвешенной мультирекомбинацией и самоадаптацией интенсивности мутации

В общем случае моделирование ЭМС БА КА включает учет как внешних воздействий (электромагнитное поле Земли и Солнца, ионосферная плазма, излучение других КА и с поверхности Земли), так и взаимовлияний в различных конструктивных элементах БА. Данная работа посвящена вопросам моделирования взаимовлияний в конструктивных элементах БА КА путем применения моделей и алгоритмов моделирования и оптимизации ЭМС. Внешние воздействия на БА не рассматриваются, однако реализованные модели позволяют учесть такие воздействия в виде источников сигнала помехи и точечных источников электромагнитного излучения.

Под оптимизацией ЭМС БА КА будем понимать поиск оптимальных параметров моделей конструктивных элементов БА по критерию обеспечения ЭМС. Большое число параметров таких моделей ведет к появлению сложных целевых функций, которые при этом не имеют аналитического представления.

Предлагается усовершенствованный алгоритм эволюционной стратегии со взвешенной мультирекомбинацией и самоадаптацией интенсивности мутации [98, 99]. Его анализ проведем для сферической модели целевой функции (1.3).

Взвешенную мультирекомбинационную эволюционную стратегию можно описать следующим образом. На первом шаге генерируется λ новых потомков в соответствии с правилом

$$\forall l = 1, \dots, \lambda: y_l \leftarrow y_p + \sigma_l \mathbf{N}_l(\mathbf{0}, \mathbf{I}), \quad (2.1)$$

где σ_l определяется механизмом адаптации интенсивности мутации. В частности, в качестве механизма адаптации для взвешенной мультирекомбинационной эволюционной стратегии можно использовать кумулятивную адаптацию длины шага [100]. Для эволюционной стратегии с кумулятивной адаптацией длины шага $\sigma_l = \sigma$, то есть этот механизм адаптации использует только одну величину мутации для всех особей в популяции.

После этого взвешенная мультирекомбинационная эволюционная стратегия рассчитывает значения целевой функции $F(y)$, упорядочивает созданные особи-потомки в соответствии с рассчитанными значениями и вычисляет взвешенную сумму

$$\langle \mathbf{z} \rangle_\omega = \sum_{l=1}^{\lambda} \omega_{l,\lambda} \mathbf{z}^{(l;\lambda)} \quad (2.2)$$

векторов \mathbf{z} . Верхний индекс $(l; \lambda)$ обозначает l -го лучшего из λ потомков, то есть особь с l -м наименьшим значением целевой функции в случае минимизации. Весовые коэффициенты $\omega_{l,\lambda}$ зависят от положения особи в упорядоченном наборе всех особей-потомков [68].

Новая текущая точка поиска вычисляется как

$$y_p \leftarrow y_p + \langle \sigma \rangle \langle \mathbf{z} \rangle_\omega, \quad (2.3)$$

где $\langle \sigma \rangle = \sigma$ для эволюционной стратегии с кумулятивной адаптацией длины шага.

Чтобы подобрать оптимальный набор весов $\omega_{i,\lambda}$, необходимо установить критерий оптимальности взвешенной мультирекомбинационной эволюционной стратегии. Зададим данный критерий как ожидаемое для следующего поколения по сравнению с предыдущим изменение значения целевой функции $F(y)$, которое обозначим как *функцию приращения качества* Δ . Таким образом, оптимально выбранные весовые коэффициенты $\omega_{i,\lambda}$ – это весовые коэффициенты, которые обеспечивают максимум функции приращения качества для взвешенной мультирекомбинационной эволюционной стратегии.

Обозначим взвешенную мультирекомбинационную эволюционную стратегию с оптимально выбранными весовыми коэффициентами $\omega_{i,\lambda}$ как $(\lambda)_{\text{opt}}$ -эволюционную стратегию.

Взвешенная мультирекомбинационная эволюционная стратегия с σ -самоадаптацией ($(\lambda)_{\text{opt}}$ - σ СА-ЭС) – это результат внедрения взвешенной мультирекомбинации в алгоритм $(\mu/\mu_I, \lambda)$ - σ СА-ЭС. Для того чтобы в данном случае можно было использовать опубликованные ранее теоретические формулы для $(\mu/\mu_I, \lambda)$ - σ СА-ЭС [101], составим алгоритм $(\lambda)_{\text{opt}}$ - σ СА-ЭС таким образом, чтобы внедрение взвешенной мультирекомбинации вносило минимум изменений в исходный алгоритм $(\mu/\mu_I, \lambda)$ - σ СА-ЭС.

Алгоритм $(\lambda)_{\text{opt}}$ - σ СА-ЭС представлен ниже.

1. Инициализировать текущую точку поиска:

$$\sigma_p \leftarrow \sigma_{\text{init}},$$

$$y_p \leftarrow y_{\text{init}}.$$

2. Сформировать λ потомков следующим образом:

$$\forall l = 1, \dots, \lambda : \begin{cases} \tilde{\sigma}_l \leftarrow \sigma_p e^{\tau N_l(0,1)}; \\ \tilde{\mathbf{z}}_l \leftarrow \mathbf{N}_l(\mathbf{0}, \mathbf{I}); \\ \tilde{\mathbf{y}}_l \leftarrow \mathbf{y}_p + \tilde{\sigma}_l \tilde{\mathbf{z}}_l; \\ \tilde{\mathbf{F}}_l \leftarrow F(\tilde{\mathbf{y}}_l). \end{cases}$$

3. Упорядочить λ потомков в соответствии с их значениями целевой функции.

4. Вычислить взвешенную сумму $\langle \mathbf{z} \rangle_\omega$ векторов $\tilde{\mathbf{z}}_l$ в соответствии с формулой

$$\langle \mathbf{z} \rangle_\omega = \sum_{l=1}^{\lambda} \omega_{l;\lambda} \tilde{\mathbf{z}}_{l;\lambda}.$$

5. Вычислить среднее значение величин мутаций по формуле

$$\langle \sigma \rangle = \frac{1}{\mu} \sum_{m=1}^{\mu} \tilde{\sigma}_{m;\lambda}.$$

6. Вычислить новую текущую точку поиска:

$$\sigma_p \leftarrow \langle \sigma \rangle,$$

$$\mathbf{y}_p \leftarrow \mathbf{y}_p + \sigma_p \langle \mathbf{z} \rangle_\omega.$$

7. Перейти к шагу 2, пока условие завершения работы алгоритма не будет выполняться.

Сравнивая данный алгоритм с $(\mu/\mu_I, \lambda)$ -сСА эволюционной стратегией, можно отметить добавление шага для вычисления взвешенной суммы $\langle \mathbf{z} \rangle_\omega$. Взвешенная сумма $\langle \mathbf{z} \rangle_\omega$ используется для расчета новой текущей точки поиска вместо среднего значения векторов параметров $\langle \tilde{\mathbf{y}} \rangle$ в исходном алгоритме $(\mu/\mu_I, \lambda)$ -ЭС.

При этом не было внесено никаких изменений в механизм σ -самоадаптации, что позволяет использовать для анализа $(\lambda)_{\text{opt}}$ - σ СА эволюционной стратегии формулы, выведенные в [101] для $(\mu/\mu_I, \lambda)$ - σ СА эволюционной стратегии. Кроме того, при описании конкретных $(\lambda)_{\text{opt}}$ - σ СА эволюционных стратегий в дальнейшем числом μ будем обозначать количество потомков, участвующих в расчете среднего значения интенсивности мутации $\langle \sigma \rangle$.

2.2. Теоретический анализ эволюционных стратегий

Обозначим символом $r^{(g)}$ расстояние между центроидом g -го поколения $\langle \mathbf{y} \rangle^{(g)}$ и оптимумом $\hat{\mathbf{y}}$, то есть $r^{(g)} = \|\mathbf{r}^{(g)}\| = \|\langle \mathbf{y} \rangle^{(g)} - \hat{\mathbf{y}}\|$; символом $s^{(g)} = \langle \sigma^{(g)} \rangle$ – среднее арифметическое родительских величин мутаций σ .

Ожидаемое для следующего поколения по сравнению с предыдущим изменение расстояния $r^{(g)}$ – *скорость прогресса* – определяется выражением

$$\varphi(s^{(g)}, r^{(g)}) = \mathbb{E} \left[r^{(g)} - r^{(g+1)} \mid (s^{(g)}, r^{(g)}) \right]. \quad (2.4)$$

Скорость прогресса близка к значению функции приращения качества при условии соответствующей нормировки, как описано ниже.

Аналогично ожидаемое для следующего поколения по сравнению с предыдущим изменение интенсивности мутации – *отклик механизма самоадаптации* – рассчитывается по формуле

$$\psi(s^{(g)}, r^{(g)}) = \mathbb{E} \left[\frac{s^{(g+1)} - s^{(g)}}{s^{(g)}} \mid (s^{(g)}, r^{(g)}) \right]. \quad (2.5)$$

Чтобы получить переменные, не зависящие от положения особей в пространстве поиска, используем нормировку

$$\varphi^* = \varphi \frac{N}{r^{(g)}} \quad (2.6)$$

и

$$s^{*(g)} = s^{(g)} \frac{N}{r^{(g)}}. \quad (2.7)$$

Посредством ряда упрощений (условие $\tau \rightarrow 0$, асимптотическое поведение при $N \rightarrow \infty$, расширение нижнего предела интегрирования до $-\infty$, применение разложения в ряд Тейлора) получаем приближенную формулу для отклика механизма самоадаптации [101]:

$$\psi(s^{*(g)}) \approx \tau^2 \left[\frac{1}{2} + e_{\mu,\lambda}^{1,1} - s^{*(g)} c_{\mu/\mu,\lambda} \right]. \quad (2.8)$$

Так называемые обобщенные коэффициенты прогресса $e_{\mu,\lambda}^{\alpha,\beta}$ рассчитываются по формуле

$$e_{\mu,\lambda}^{\alpha,\beta} = \frac{\lambda - \mu}{\sqrt{2\pi}^{\alpha+1}} \binom{\lambda}{\mu} \int_{-\infty}^{\infty} t^\beta e^{-\frac{\alpha+1}{2}t^2} \Phi(t)^{\lambda-\mu-1} (1 - \Phi(t))^{\mu-\alpha} dt, \quad (2.9)$$

где $\alpha, \beta, \mu, \lambda$ – натуральные числа. Коэффициент прогресса $c_{\mu/\mu,\lambda}$ является частным случаем обобщенных коэффициентов прогресса:

$$c_{\mu/\mu,\lambda} = e_{\mu,\lambda}^{1,0}. \quad (2.10)$$

2.3. Анализ мультирекомбинационной эволюционной стратегии

Рассмотрим суммирование вектора мутаций $\sigma^{(g)}\mathbf{z}$ и текущего центроида родительской популяции $\langle \mathbf{y} \rangle^{(g)}$ (рис. 2.1). Результат суммирования – особь-потомок с вектором параметров $\mathbf{y}^{(g+1)}$. Принимая соответствующие расстояния от $\langle \mathbf{y} \rangle^{(g)}$ и $\mathbf{y}^{(g+1)}$ до оптимизатора за $r^{(g)}$ и $r^{(g+1)}$, получим разницу $\delta(\sigma^{(g)}\mathbf{z}) = (r^{(g)})^2 - (r^{(g+1)})^2$ между значениями целевой функции $F(\langle \mathbf{y} \rangle^{(g)}) = (r^{(g)})^2$ и $F(\mathbf{y}^{(g+1)}) = (r^{(g+1)})^2$, являющуюся *приростом целевой функции*, который соответствует вектору $\sigma^{(g)}\mathbf{z}$.

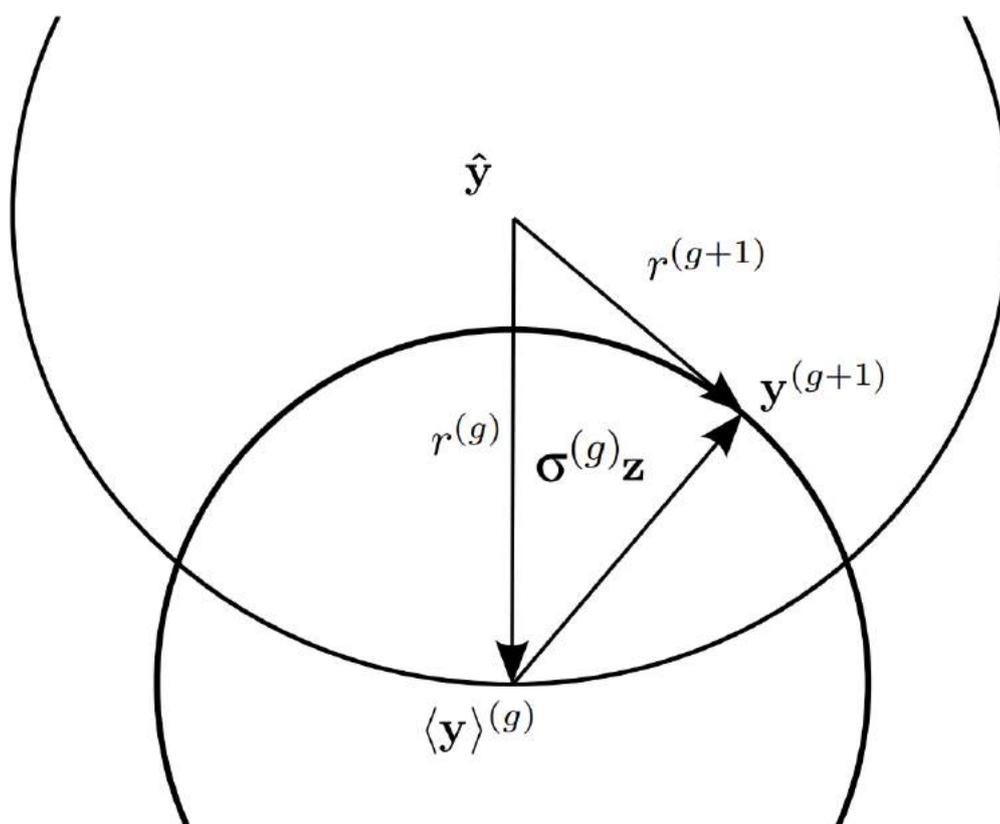


Рис. 2.1. Сложение вектора мутаций $\sigma^{(g)}\mathbf{z}$ и текущего центроида родительской популяции $\langle \mathbf{y} \rangle^{(g)}$ для получения потомка с вектором параметров $\mathbf{y}^{(g+1)}$

Чтобы получить формулу прироста целевой функции, не зависящую от положения особей в пространстве поиска, используем нормировку

$$\delta^* = \delta \frac{N}{2 (r^{(g)})^2}. \quad (2.11)$$

Тогда нормированная функция приращения качества мультирекомбинационной эволюционной стратегии выражается через нормированный прирост целевой функции как [68]

$$\Delta^* = -\frac{N}{2} \mathbb{E} \left[\log \left(1 - \frac{2}{N} \delta^* (\langle \mathbf{z} \rangle) \right) \right]. \quad (2.12)$$

Посредством ряда упрощений (учет асимптотического поведения при $N \rightarrow \infty$, предположение, что нормированная интенсивность мутации σ^* имеет порядок $\mathcal{O}(1)$, применение разложения в ряд Тейлора) получаем приближенную формулу для нормированной функции приращения качества [68]

$$\Delta^* \stackrel{N \rightarrow \infty}{\cong} \mathbb{E} \left[\delta^* (\langle \sigma^{*(g)} \mathbf{z} \rangle) \right]. \quad (2.13)$$

Определив ожидаемое значение прироста целевой функции, запишем формулу нормированной функции приращения качества для взвешенной мультирекомбинационной эволюционной стратегии [68]

$$\Delta^* \stackrel{N \rightarrow \infty}{\cong} \sigma^{*(g)} \sum_{k=1}^{\lambda} \omega_{k,\lambda} E_{k,\lambda} - \frac{(\sigma^{*(g)})^2}{2} \sum_{k=1}^{\lambda} \omega_{k,\lambda}^2, \quad (2.14)$$

где

$$E_{k,\lambda} = \int_{-\infty}^{\infty} x p_{k,\lambda}(x) dx \quad (2.15)$$

является математическим ожиданием $(\lambda + 1 - k)$ -й порядковой статистики, а функция плотности распределения вероятности имеет вид

$$p_{k,\lambda}(x) = \frac{1}{\sqrt{2\pi}} \frac{\lambda!}{(\lambda - k)!(k - 1)!} e^{-\frac{1}{2}x^2} [\Phi(x)]^{\lambda-k} [1 - \Phi(x)]^{k-1}. \quad (2.16)$$

Подставив формулу (2.16) в формулу (2.15), видим, что $E_{k,\lambda}$ – частный случай обобщенных коэффициентов прогресса:

$$E_{k,\lambda} = e_{k-1,\lambda}^{0,1}. \quad (2.17)$$

Очевидно, что величина функции приращения качества (2.14) максимальна при

$$\omega_{k,\lambda} = \frac{E_{k,\lambda}}{\sigma^*(g)}. \quad (2.18)$$

Оптимальные, обеспечивающие максимальное значение функции приращения качества весовые коэффициенты (2.18) зависят от нормированной интенсивности мутации, которая изменяется в процессе работы эволюционной стратегии. Поэтому требуется непрерывная адаптация весовых коэффициентов после запуска эволюционной стратегии. Вместо этого $(\lambda)_{\text{opt}}$ -ЭС использует фиксированные весовые коэффициенты

$$\omega_{k,\lambda} = E_{k,\lambda} \text{ for } k = 1, \dots, \lambda, \quad (2.19)$$

а механизм адаптации интенсивности мутации выбирает величину мутации таким образом, что набор весов (2.19) является всегда оптимальным [68].

Нормированную функцию приращения качества $(\lambda)_{\text{opt}}$ -ЭС с оптимальными весовыми коэффициентами (2.19) можно записать с помощью выражения (2.14) в виде

$$\Delta^*(\sigma^{*(g)}) = W_\lambda \left(\sigma^{*(g)} - \frac{(\sigma^{*(g)})^2}{2} \right), \quad (2.20)$$

где

$$W_\lambda = \sum_{k=1}^{\lambda} E_{k,\lambda}^2.$$

Величина функции приращения качества (2.20) максимальна при $\sigma^{*(g)} = 1$.

Выбор весов (2.19) не единственно возможный. Аналогично можно выбрать [68]

$$\omega_{k,\lambda} = \frac{E_{k,\lambda}}{\kappa}, \quad k = 1, \dots, \lambda, \quad \kappa > 0. \quad (2.21)$$

В этом случае нормированная оптимальная интенсивность мутации $\sigma^{*(g)} = \kappa$, и при адаптации интенсивности мутации могут быть получены большие значения оптимальной интенсивности мутации $\sigma^{*(g)}$ по сравнению с выбором весов (2.19). Хотя это не приводит к увеличению максимально возможного значения функции приращения качества в рассматриваемом случае, когда шум не влияет на результат вычисления целевой функции, этот факт может быть полезен в пространствах поиска, подверженных воздействию шума, так как эволюционная стратегия будет работать с большими значениями интенсивности мутации, позволяющими увеличить отношение сигнал-шум [11].

Важно отметить, что формула (2.20) получена для $(\lambda)_{\text{opt}}$ -ЭС, поэтому рассматривается общая нормированная интенсивность мутации $\sigma^{(g)}$ для каждого поколения g . Это неверно для $(\lambda)_{\text{opt}}$ -сСА-ЭС, так как каждая особь-потомок имеет собственную величину мутации, определяемую механизмом самоадаптации. Поэтому требуется доказать, что

нормированное среднее значение $s^{(g)}$ может быть использовано в формуле (2.20) вместо $\sigma^{(g)}$ при условии $N \rightarrow \infty$. Действительно, так как

$$s^{(g)} = \langle \sigma^{(g)} \rangle = \frac{1}{\mu} \sum_{m=1}^{\mu} \sigma_{m;\lambda}^{(g)} = \frac{1}{\mu} \sum_{m=1}^{\mu} \sigma_p^{(g)} e^{\tau N_m(0,1)}$$

и

$$\tau = \frac{1}{\sqrt{2N}} \stackrel{N \rightarrow \infty}{\equiv} 0,$$

то

$$s^{(g)} \stackrel{N \rightarrow \infty}{\equiv} \frac{1}{\mu} \sum_{m=1}^{\mu} \sigma_p^{(g)} \cdot 1 = \sigma_p^{(g-1)}. \quad (2.22)$$

Формула (2.22) показывает, что среднее значение $s^{(g)}$ величин мутации, которые созданы с использованием логарифмически нормального оператора на шаге 2 алгоритма $(\lambda)_{\text{opt}}\text{-}\sigma\text{CA}$ эволюционной стратегии, стремится к «родительской» (соответствующей текущей точке поиска) величине мутации $\sigma_p^{(g-1)}$ при условии $N \rightarrow \infty$. Это доказательство применимо и к нормированным величинам. Таким образом, обосновано использование $s^{(g)}$ в формуле (2.20) вместо $\sigma^{(g)}$.

Покажем, что нормированная функция приращения качества (2.13) в квадратичном сферическом пространстве поиска близка по значению к нормированной скорости прогресса $\varphi^* = \mathbb{E} \left[\frac{N}{r^{(g)}} (r^{(g)} - r^{(g+1)}) \right]$ при условии $N \rightarrow \infty$ [102].

Чтобы нормированная интенсивность мутации имела конечное значение, нормированный прирост целевой функции должен быть конечен. Поэтому с ростом N ожидаемое значение разницы между расстоянием $r^{(g)}$ родительского центроида до оптимума и расстоянием $r^{(g+1)}$ выбранного потомка до оптимума стремится к нулю. Действительно, так как

$$\begin{aligned}
\Delta^* \stackrel{N \rightarrow \infty}{\equiv} \mathbb{E} \left[\delta^* \left(s^{*(g)} \langle \mathbf{z} \rangle \right) \right] &= \mathbb{E} \left[\frac{N}{2 (r^{(g)})^2} \delta \left(s^{*(g)} \langle \mathbf{z} \rangle \right) \right] = \\
&= \mathbb{E} \left[\frac{N}{2 (r^{(g)})^2} \left((r^{(g)})^2 - (r^{(g+1)})^2 \right) \right] = \\
&= \mathbb{E} \left[N \frac{r^{(g)} - r^{(g+1)}}{r^{(g)}} \frac{r^{(g)} + r^{(g+1)}}{2r^{(g)}} \right] \quad (2.23)
\end{aligned}$$

и

$$\frac{r^{(g)} + r^{(g+1)}}{2r^{(g)}} \rightarrow 1,$$

то

$$\Delta^* \left(s^{*(g)} \right) \stackrel{N \rightarrow \infty}{\equiv} \mathbb{E} \left[\frac{N}{r^{(g)}} \left(r^{(g)} - r^{(g+1)} \right) \cdot 1 \right] = \varphi^* \left(s^{*(g)} \right). \quad (2.24)$$

2.4. Анализ $(\mu/\mu_I, \lambda)$ -сСА эволюционной стратегии

В случае корректно работающей σ -самоадаптации ожидаемое значение нормированной интенсивности мутации должно со временем войти в стационарное состояние $\lim_{g \rightarrow \infty} s^{*(g)} = s_{\text{st}}^*$. С помощью детерминистского уравнения [101]

$$s^{*(g+1)} = s^{*(g)} \frac{1 + \psi(s^{*(g)}, N)}{1 - \frac{\varphi^*(s^{*(g)}, N)}{N}} \quad (2.25)$$

получаем условие стационарного состояния

$$\frac{\varphi^*(s_{\text{st}}^*)}{N} = -\psi \left(s_{\text{st}}^* \right), \quad (2.26)$$

которое можно использовать для вывода формулы стационарной интенсивности мутации. Так как

$\varphi^*(s^{*(g)}) = \Delta^*(s^{*(g)})$, что верно для квадратичного сферического пространства поиска при условии бесконечной размерности пространства поиска $N \rightarrow \infty$, используем выражение (2.20) вместо $\varphi^*(s^{*(g)})$ в равенстве (2.26). Подставив (2.8) и (2.20) в (2.26), получим

$$W_\lambda \left(s_{st}^* - \frac{(s_{st}^*)^2}{2} \right) = -\alpha^2 \left[\frac{1}{2} + e_{\mu,\lambda}^{1,1} - s_{st}^* c_{\mu/\mu,\lambda} \right], \quad (2.27)$$

где параметр самообучения τ в формуле (2.8) выбран в соответствии с равенством

$$\tau = \frac{\alpha}{\sqrt{N}}. \quad (2.28)$$

С учетом равенства (2.28) формула (2.8) становится независимой от размерности пространства поиска N .

Как видим, стационарное состояние зависит от параметра α . Рассмотрим предельные случаи. При $\alpha \rightarrow \infty$, для того чтобы функция приращения качества имела конечное значение в правой части формулы (2.27), функция отклика механизма самоадаптации должна стремиться к нулю быстрее, чем $\alpha \rightarrow \infty$. Таким образом, стационарное состояние определяется при условии $\alpha \rightarrow \infty$ нулем функции отклика механизма самоадаптации

$$s_{\psi_0}^* = \frac{0,5 + e_{\mu,\lambda}^{1,1}}{c_{\mu/\mu,\lambda}}. \quad (2.29)$$

При $\alpha \rightarrow 0$ левая часть формулы (2.27) должна стремиться к нулю. Стационарное состояние в этом случае определяется вторым нулем функции приращения качества

$$s_{\Delta_0}^* = 2, \quad (2.30)$$

тогда как тривиальный первый нуль функции приращения качества $s_{\Delta_0}^* = 0$ не может быть достигнут правильно работающей эволюционной стратегией, пока текущая точка поиска не приблизится к оптимуму. Таким образом, параметр α управляет стационарным состоянием интенсивности мутации.

Получим аналитическую формулу s_{st}^* . Решение квадратного уравнения (2.27) по переменной s_{st}^* дает

$$s_{st}^* = 1 - \frac{c_{\mu/\mu,\lambda}\alpha^2}{W_\lambda} + K, \quad (2.31)$$

где

$$K = \sqrt{1 + (1 - 2c_{\mu/\mu,\lambda} + 2e_{\mu,\lambda}^{1,1})\frac{\alpha^2}{W_\lambda} + \frac{c_{\mu/\mu,\lambda}^2\alpha^4}{W_\lambda^2}}. \quad (2.32)$$

Результаты вычислений по формуле (2.31) показаны на рис. 2.2.

Принимая во внимание равенство $\varphi^*(s^{*(g)}) = \Delta^*(s^{*(g)})$ и подставляя стационарную величину мутации (2.31) в функцию приращения качества (2.20), получаем стационарную скорость прогресса

$$\varphi_{st}^*(\alpha) = \frac{W_\lambda}{2} \left[1 - \left(\frac{c_{\mu/\mu,\lambda}\alpha^2}{W_\lambda} - K \right)^2 \right]. \quad (2.33)$$

Результаты вычислений по формуле (2.33) показаны на рис. 2.3. Рассмотрим общий характер зависимостей для стационарной интенсивности мутации и стационарной скорости прогресса.

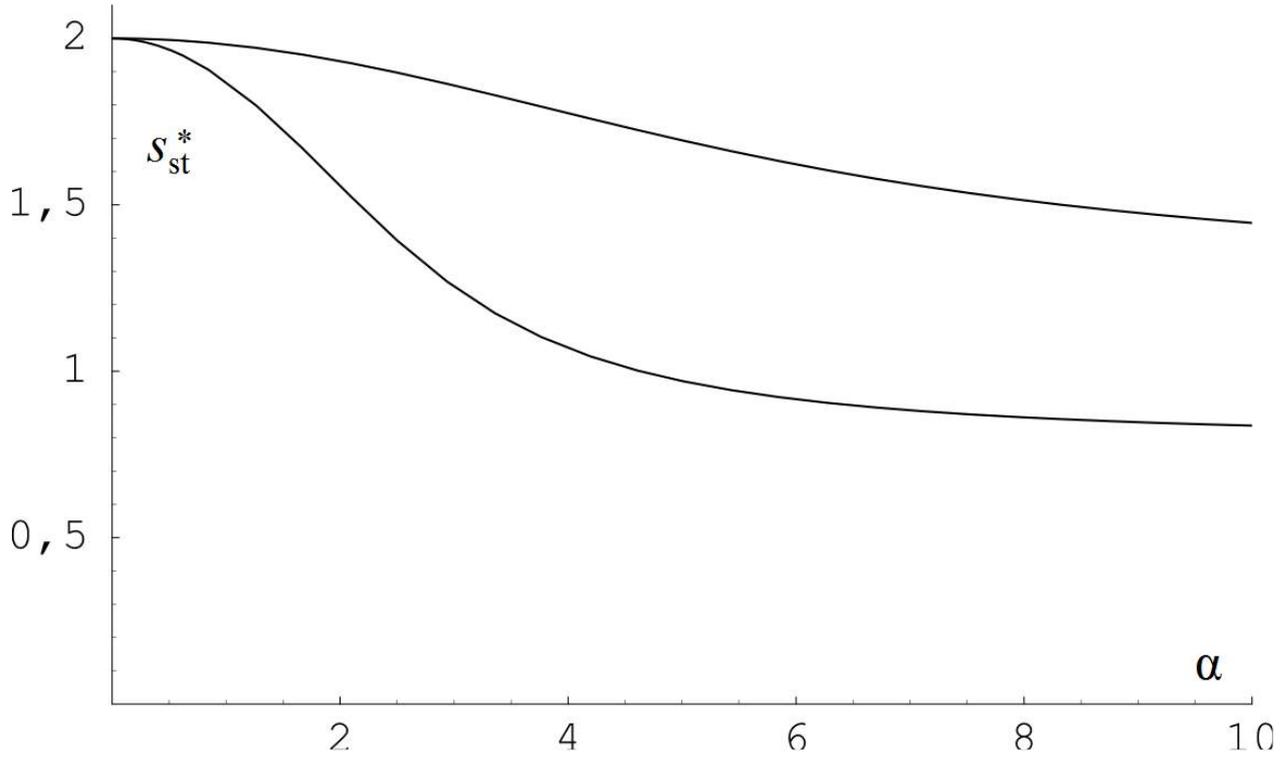


Рис. 2.2. Стационарная интенсивность мутации взвешенной мультирекомбинационной эволюционной стратегии с σ -самоадаптацией как функция параметра α : нижняя кривая для $\mu = 4$, $\lambda = 10$, верхняя – для $\mu = 10$, $\lambda = 60$

Величина функции приращения качества (2.20), записанная в виде $\Delta^*(s^{*(g)}) = W_\lambda(s^{*(g)} - (s^{*(g)})^2/2)$, принимает оптимальное значение $\Delta_{\max}^* = W_\lambda/2$ при $s_{\Delta_{\max}}^* = 1$. Для получения оптимального прогресса в стационарном состоянии представляется наиболее выгодным условие $s_{\text{st}}^* = s_{\Delta_{\max}}^* = 1$, что ведет к соотношению

$$1 - \frac{c_{\mu/\mu,\lambda}\alpha_{\text{opt}}^2}{W_\lambda} + \sqrt{1 + (1 - 2c_{\mu/\mu,\lambda} + 2e_{\mu,\lambda}^{1,1})\frac{\alpha_{\text{opt}}^2}{W_\lambda} + \frac{c_{\mu/\mu,\lambda}^2\alpha_{\text{opt}}^4}{W_\lambda^2}} = 1,$$

откуда

$$\alpha_{\text{opt}} = \sqrt{\frac{W_\lambda}{2c_{\mu/\mu,\lambda} - 2e_{\mu,\lambda}^{1,1} - 1}} \stackrel{\lambda \geq 50}{\approx} \begin{cases} 3\sqrt{\lambda} & \mu/\lambda = 0,3 \\ 1,5\sqrt{\lambda} & \mu/\lambda = 0,4 \end{cases}. \quad (2.34)$$

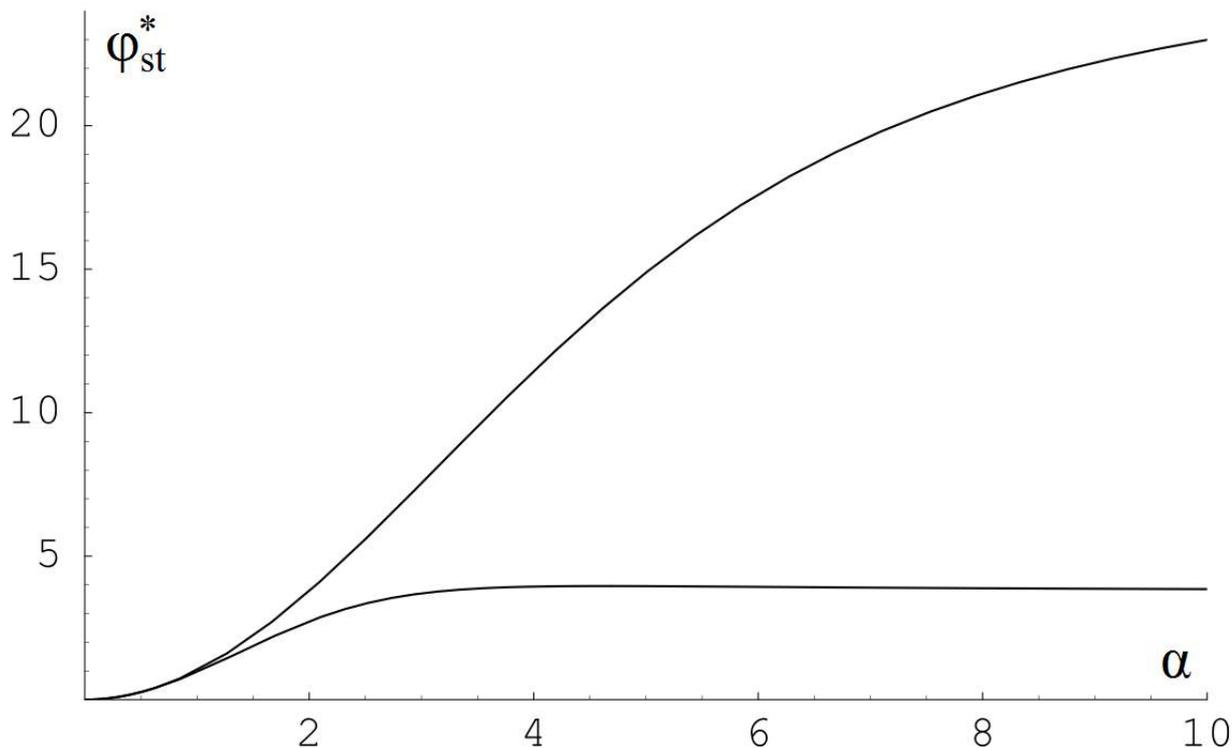


Рис. 2.3. Стационарная скорость прогресса взвешенной мультирекомбинационной эволюционной стратегии с σ -самоадаптацией как функция параметра α : нижняя кривая для $\mu = 4$, $\lambda = 10$, верхняя – для $\mu = 10$, $\lambda = 60$

2.5. Пространства поиска конечной размерности

Вывод формул стационарного состояния сделан с учетом бесконечной размерности пространства поиска. Эти формулы получены с помощью упрощений, описанных в подразделе 2.2. Для конечномерного пространства поиска необходима проверка представленных выше выкладок. Проведем численные эксперименты в пространствах поиска конечной размерности.

Сравнение результатов вычислений по формуле (2.33) с результатами экспериментов приведено на рис. 2.4. Если не указано иное, используется исходный вектор $y^{(0)} = 1000$ и исходная интенсивность мутации $\sigma^{(0)} = 10$. В экспери-

ментах взвешенная мультирекомбинационная эволюционная стратегия с σ -самоадаптацией работает с числом родителей $\mu = 4$ и числом потомков $\lambda = 10$. Сбор данных начинается с 2000-го поколения и выполняется до 4000-го поколения. Для расчета стационарной скорости прогресса используется формула [11]

$$\Phi_{st}^* = \frac{N}{G} \ln \left(\frac{r^{(g_0)}}{r^{(g)}} \right), \quad (2.35)$$

где $G = g - g_0$, $g_0 = 2000$, $g = 4000$.

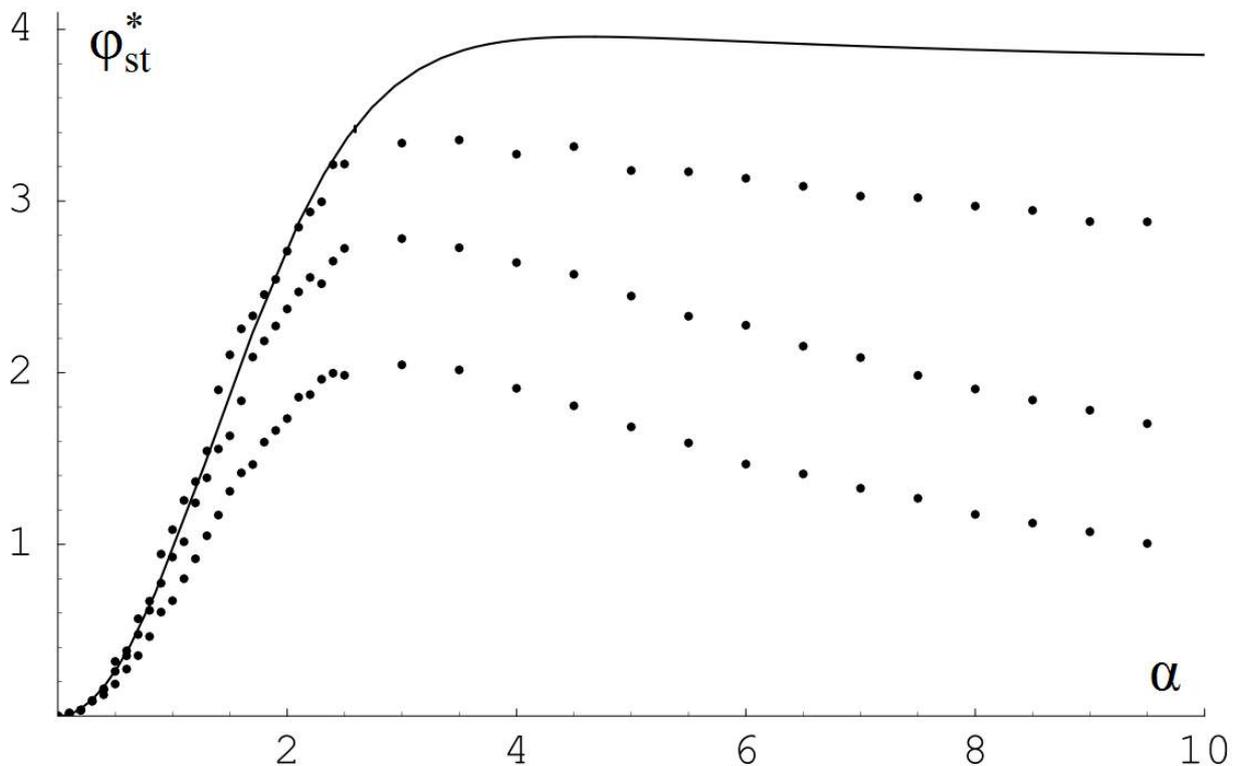


Рис. 2.4. Стационарная скорость прогресса взвешенной мультирекомбинационной эволюционной стратегии с σ -самоадаптацией ($\mu = 4$, $\lambda = 10$) как функция параметра α : сплошная кривая – результаты вычислений; точки – результаты экспериментов, усредненные по 10 запускам (приведены снизу вверх результаты экспериментов для $N = 30, 100, 1000$)

Эксперименты показывают, что нормированная стационарная скорость прогресса близка к теоретической кривой $N \rightarrow \infty$ для малых значений параметра α . Рассмотрим фиксированное значение размерности пространства поиска N . Для больших значений параметра α наблюдается уменьшение экспериментальных значений нормированной стационарной скорости прогресса по причине роста параметра самообучения τ . Большие значения параметра самообучения могут сократить время, затрачиваемое эволюционной стратегией на самоадаптацию, так как они позволяют получить больший отклик механизма самоадаптации для заданных значений s^* . Однако увеличение параметра самообучения также несколько уменьшает максимально достижимую скорость прогресса [11].

Сравнение результатов вычислений по формуле (2.33) для различных значений параметра самообучения τ с результатами экспериментов для $N = 1000, 100, 30$ приведено на рис. 2.5. Эксперименты показывают, что нормированная стационарная скорость прогресса увеличивается с ростом параметра самообучения τ до достижения максимальной точки, а затем уменьшается. Как максимальная нормированная стационарная скорость прогресса, так и оптимальное значение параметра τ зависит от размерности пространства поиска.

2.6. Сравнительный анализ $(\mu/\mu_I, \lambda)$ и $(\lambda)_{opt}$ эволюционных стратегий

Стационарная скорость прогресса для самоадаптивной $(\mu/\mu_I, \lambda)$ -ЭС задается формулой [101]

$$\varphi_{st}^*(\alpha) = \alpha^2 \left(\mu c_{\mu/\mu, \lambda} (1 - \alpha^2) + c_{\mu/\mu, \lambda} K - \frac{1}{2} - e_{\mu, \lambda}^{1,1} \right), \quad (2.36)$$

где

$$K = \sqrt{\mu^2 c_{\mu/\mu, \lambda}^2 (1 - \alpha^2)^2 + 2\mu\alpha^2 \left(\frac{1}{2} + e_{\mu, \lambda}^{1,1} \right)}.$$

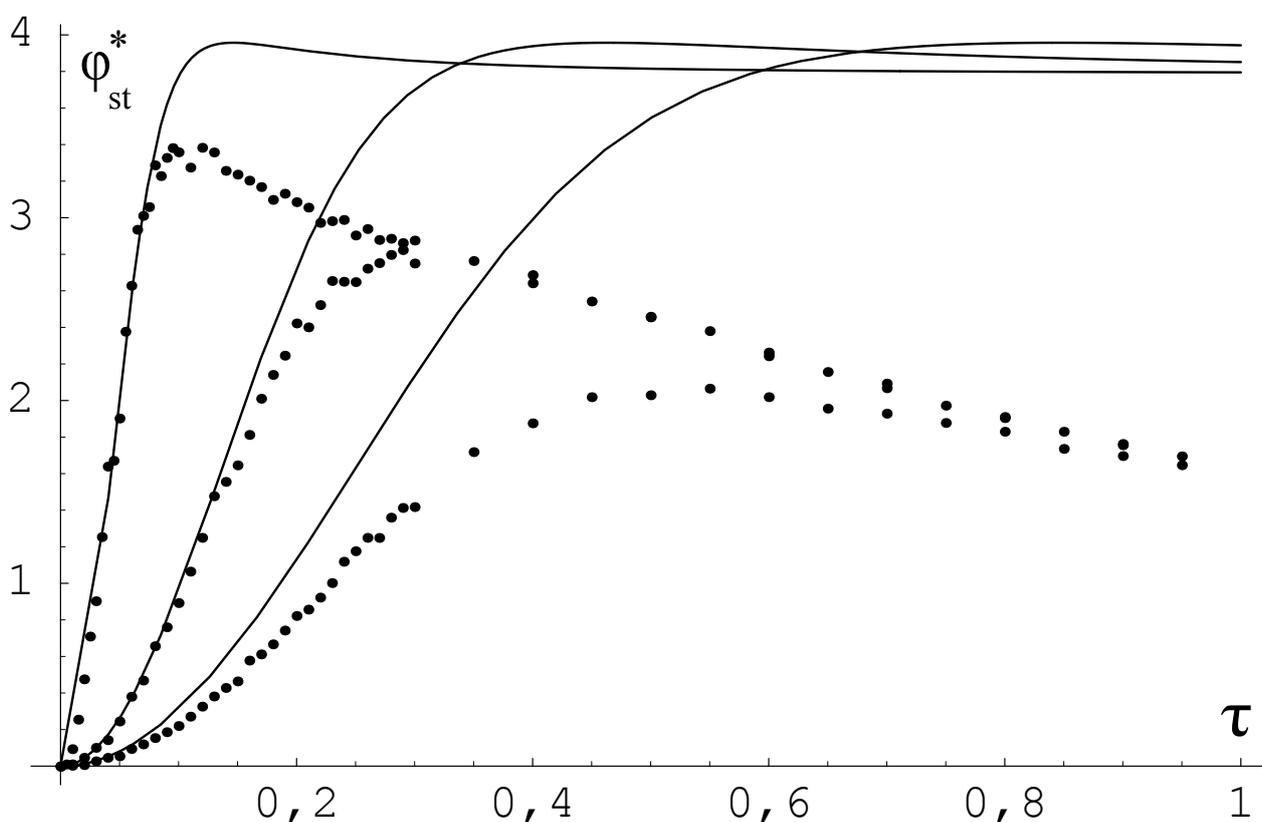


Рис. 2.5. Стационарная скорость прогресса взвешенной мультирекомбинационной эволюционной стратегии с σ -самоадаптацией ($\mu = 4, \lambda = 10$) как функция параметра самообучения τ : слева направо результаты для $N = 1000, 100, 30$; сплошные кривые – результаты вычислений; точки – результаты экспериментов, усредненные по 10 запускам

$(\lambda)_{\text{opt}}$ -ЭС с кумулятивной адаптацией длины шага использует отличный от σ -самоадаптации механизм для адаптации интенсивности мутации. Взвешенные суммы $\langle \mathbf{z} \rangle_{\omega}$ накапливаются во время работы эволюционной стратегии N -размерным вектором \mathbf{l} с целью отслеживания шагов стратегии в пространстве поиска. Вектор \mathbf{l} определяется как $\mathbf{l}^{(0)} = \mathbf{0}$ и

$$\mathbf{l}^{(g+1)} = (1 - c)\mathbf{l}^{(g)} + \sqrt{\frac{c(2 - c)}{W_{\lambda}}} \langle \mathbf{z} \rangle_{\omega}^{(g)}, \quad (2.37)$$

где $c = 1/\sqrt{N}$ – параметр кумуляции [68]. Правило адаптации интенсивности мутации задается формулой

$$\sigma^{(g+1)} = \sigma^{(g)} e^{\frac{\|\mathbf{l}^{(g+1)}\|^2 - N}{2DN}}, \quad (2.38)$$

где $D = 1/c$ – параметр затухания [68].

Важно отметить, что $(\lambda)_{\text{opt}}$ -ЭС с кумулятивной адаптацией длины шага не способна достигнуть интенсивности мутации $\sigma^{*(g)} = 1$, которая использовалась для задания оптимальных весовых коэффициентов (2.19). Поэтому $(\lambda)_{\text{opt}}$ -ЭС не может достигнуть максимального значения функции приращения качества (2.20) [68]. Реальное усредненное значение функции приращения качества для $(\lambda)_{\text{opt}}$ -ЭС с кумулятивной адаптацией длины шага определяется по формуле [68]

$$\Delta_{avg}^* = (\sqrt{2} - 1) W_{\lambda}. \quad (2.39)$$

Так как $\varphi^*(s^{*(g)}) = \Delta^*(s^{*(g)})$, формулу (2.39) можно использовать для сравнения с экспериментальными результатами взвешенной мультирекомбинационной ЭС с σ -самоадаптацией.

Результаты вычислений по формулам (2.33), (2.36) и (2.39) сравниваются на рис. 2.6 с экспериментальными результатами взвешенной мультирекомбинационной ЭС с σ -самоадаптацией и кумулятивной адаптацией длины шага. Также показаны экспериментальные результаты взвешенной мультирекомбинационной ЭС с σ -самоадаптацией с набором весов

$$\omega_{k,\lambda} = \begin{cases} 1/\mu, & \text{если } 1 \leq k \leq \mu, \\ 0 & \text{иначе.} \end{cases} \quad (2.40)$$

В данном случае взвешенная мультирекомбинационная эволюционная стратегия является просто $(\mu/\mu_I, \lambda)$ - σ СА эволюционной стратегией, а точка поиска – центроидом популяции, которая состоит из μ лучших из λ созданных потомков.

На рис. 2.6 сплошная горизонтальная линия, обозначенная $(10)_{\text{opt}} + \text{КА}$, показывает результаты вычислений по формуле (2.39) для $(10)_{\text{opt}}$ -ЭС с кумулятивной адаптацией длины шага, а пунктирная горизонтальная линия – результаты экспериментов для той же ЭС, усредненные по 10 запускам. Сплошные кривые отображают результаты вычислений по формуле (2.36) для самоадаптационной $(4/4_I, 10)$ -ЭС и по формуле (2.33) для взвешенной мультирекомбинационной ЭС с σ -самоадаптацией. Точки показывают результаты экспериментов для $N = 1000$ со взвешенной мультирекомбинационной ЭС с σ -самоадаптацией, усредненные по 10 запускам. Приведены результаты для набора весов (2.40), обозначенные $(4/4, 10)$, и для стандартных весовых коэффициентов $\omega_{k,\lambda} = E_{k,\lambda}$, обозначенные $(10)_{\text{opt}} + \sigma\text{СА}$.

Сравнение $(\lambda)_{\text{opt}}\text{-}\sigma\text{CA-}\text{ЭС}$ с $(\mu/\mu_I, \lambda)\text{-}\text{ЭС}$ показывает, что взвешенная мультирекомбинационная ЭС с σ -самоадаптацией превосходит по скорости прогресса простую мультирекомбинационную ЭС, если значение параметра α достаточно большое. Такой результат объясняется тем, что $(\lambda)_{\text{opt}}\text{-}\sigma\text{CA-}\text{ЭС}$ использует информацию о взаимном упорядочении всех λ векторов \tilde{z}_l , тогда как $(\mu/\mu_I, \lambda)\text{-}\text{ЭС}$ принимает во внимание векторы \tilde{z}_l только лучших μ потомков с равными весовыми коэффициентами.

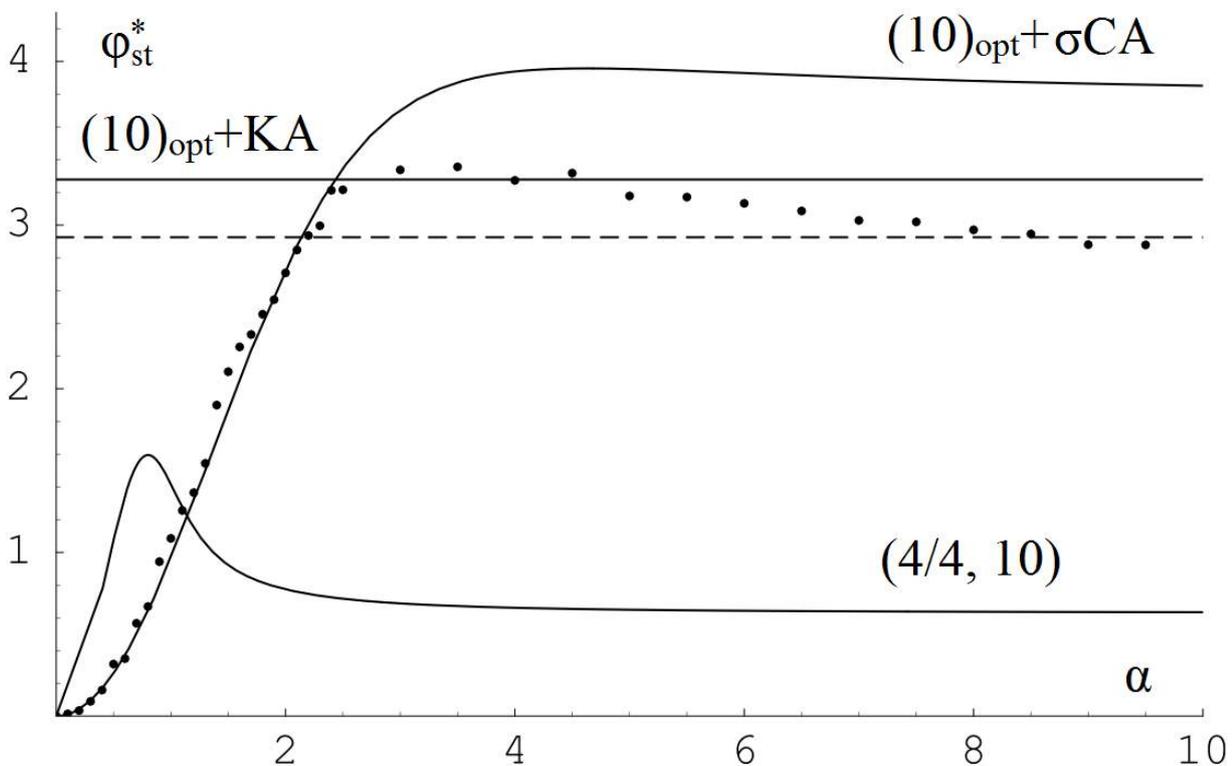


Рис. 2.6. Стационарная скорость прогресса как функция параметра α для $(10)_{\text{opt}}\text{-}\sigma\text{CA-}\text{ЭС}$ ($(10)_{\text{opt}} + \sigma\text{CA}$), самоадаптационной $(4/4_I, 10)\text{-}\text{ЭС}$ ($(4/4, 10)$) и $(10)_{\text{opt}}\text{-}\text{ЭС}$ с кумулятивной адаптацией длины шага ($(10)_{\text{opt}} + \text{KA}$)

В то же время $(\mu/\mu_I, \lambda)\text{-}\text{ЭС}$ работает лучше для малых значений параметра α , так как $(\lambda)_{\text{opt}}\text{-}\sigma\text{CA-}\text{ЭС}$ использует общий вектор мутаций $\langle z \rangle$ для всех особей в новом поколении

и, поэтому, требует для оптимальной работы больших значений параметра τ для данной размерности пространства поиска N .

Сравнение $(\lambda)_{\text{opt}}\text{-}\sigma\text{СА-ЭС}$ с $(\lambda)_{\text{opt}}\text{-ЭС}$ с кумулятивной адаптацией длины шага показывает, что взвешенная мультирекомбинационная ЭС с σ -самоадаптацией может достичь той же производительности, что и взвешенная мультирекомбинационная ЭС с кумулятивной адаптацией длины шага, если значение параметра α достаточно большое, и даже превзойти ее в определенном диапазоне значений этого параметра. Причина заключается в том, что оптимальная скорость прогресса $(\lambda)_{\text{opt}}\text{-}\sigma\text{СА-ЭС}$ может быть достигнута только для определенных значений параметра α , тогда как $(\lambda)_{\text{opt}}\text{-ЭС}$ с кумулятивной адаптацией длины шага не использует данный параметр в алгоритме.

2.7. Разработка геометрических моделей

конструктивных элементов печатных плат
и корпусов бортовой аппаратуры космических
аппаратов для квазистатического
и электродинамического анализа
электромагнитной совместимости

Для анализа ЭМС будем применять следующие модели: двух- и трехмерные квазистатические модели сложных структур проводников и диэлектриков, трехмерные электродинамические модели проводных структур. Эти алгоритмические модели накладывают требования на формат представления входных данных, который отличается от форматов данных, используемых программным обеспечением для проектирования печатных плат (ПП) и корпусов

БА КА. Таким образом, необходимо разработать геометрические модели, которые обеспечивают корректное представление исходных данных о геометрии ПП и корпусов БА КА в форме, пригодной для обработки алгоритмами квазистатического и электродинамического анализа ЭМС.

Геометрические модели печатных плат. В качестве исходных данных, описывающих геометрию ПП, использовались файлы ПО для проектирования ПП Altium Designer. На основе формата данных файлов ПО Altium Designer разработаны два типа геометрических моделей печатных плат: трехмерная геометрическая модель для квазистатического анализа (вычисление матриц параметров, рис. 2.7) и трехмерная тонкопроводная геометрическая модель для электродинамического анализа (вычисление токов, рис. 2.8).

Для примера на рис. 2.7 показана трехмерная геометрическая модель ПП радиотехнического блока, созданная из файла ПО Altium Designer. Так как трехмерная квазистатическая модель сложных структур проводников и диэлектриков [89] обрабатывает только ортогональные границы проводник-диэлектрик и диэлектрик-диэлектрик, при создании геометрических моделей ПП выполняется аппроксимация трасс, переходных отверстий и контактных площадок ПП ортогональными поверхностями. Для полученных геометрических моделей переходных отверстий и контактных площадок вычисляются матрицы емкостей и проводимостей, а для поперечных сечений регулярных отрезков – матрицы погонных емкостей, проводимостей и индуктивностей многопроводных линий передачи.

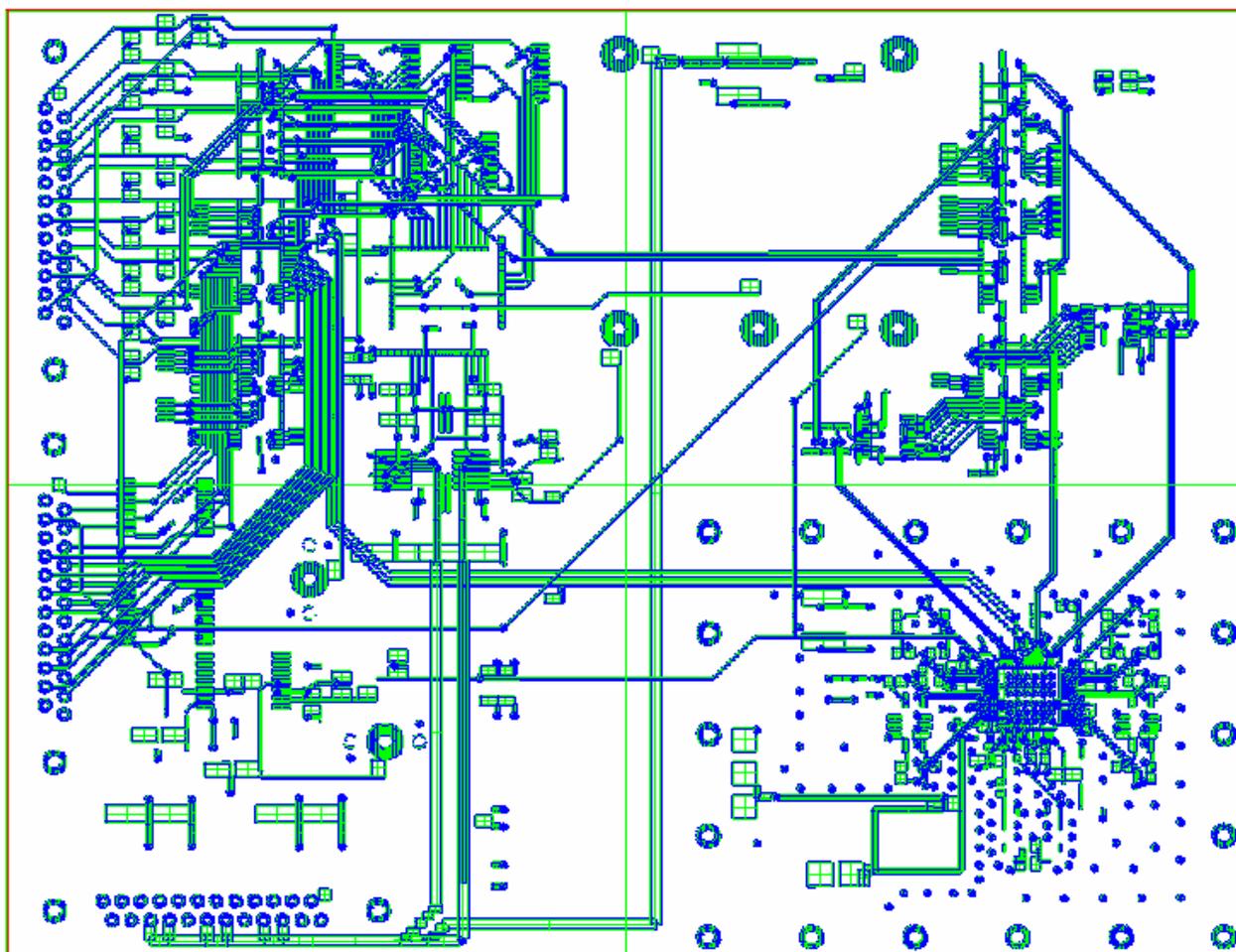


Рис. 2.7. Трехмерная геометрическая модель печатной платы

Для вычисления значений напряженности электрического поля в заданных точках ПП разработана трехмерная тонкопроводная геометрическая модель. На рис. 2.8 показан пример такой модели для ПП радиотехнического блока.

Так как трехмерная электродинамическая модель проводных структур [76] обеспечивает корректное вычисление токов для конфигураций из отрезков малого диаметра, при создании тонкопроводной модели трассы ПП аппроксимируются проводами малого диаметра. При расчете токов также предполагается, что ток протекает по оси бесконечно тонкого отрезка проводника. Затем из значений токов вычисляются значения напряженности электрического поля.

Геометрические модели корпусов. Для описания геометрии корпусов БА КА разработана трехмерная тонкопроводная геометрическая модель, при создании которой поверхности корпуса аппроксимируются сеткой из проводов (рис. 2.9). Исследование точности результатов моделирования с использованием разработанной тонкопроводной геометрической модели выполнено в разделе 3.

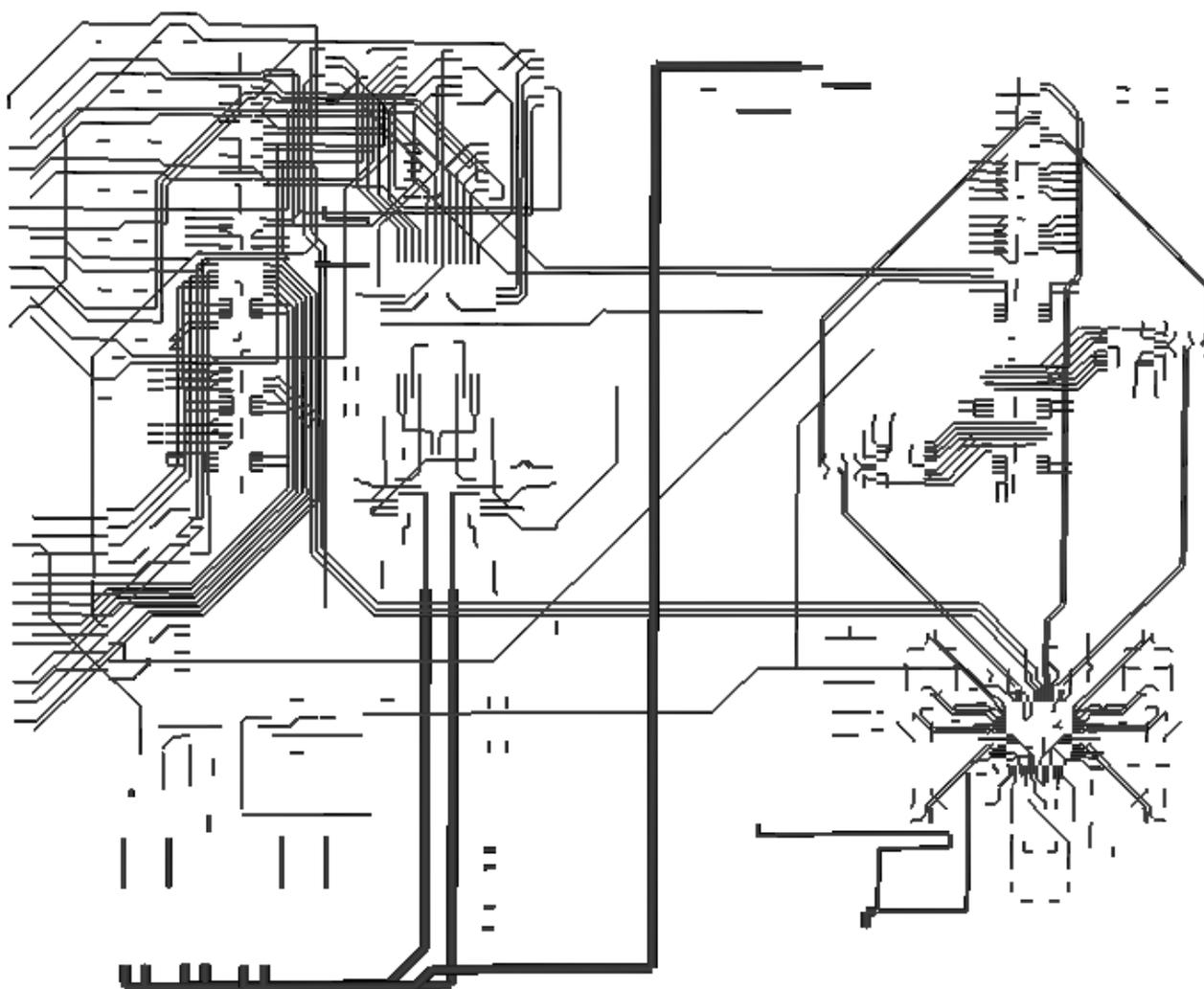


Рис. 2.8. Трехмерная тонкопроводная модель печатной платы

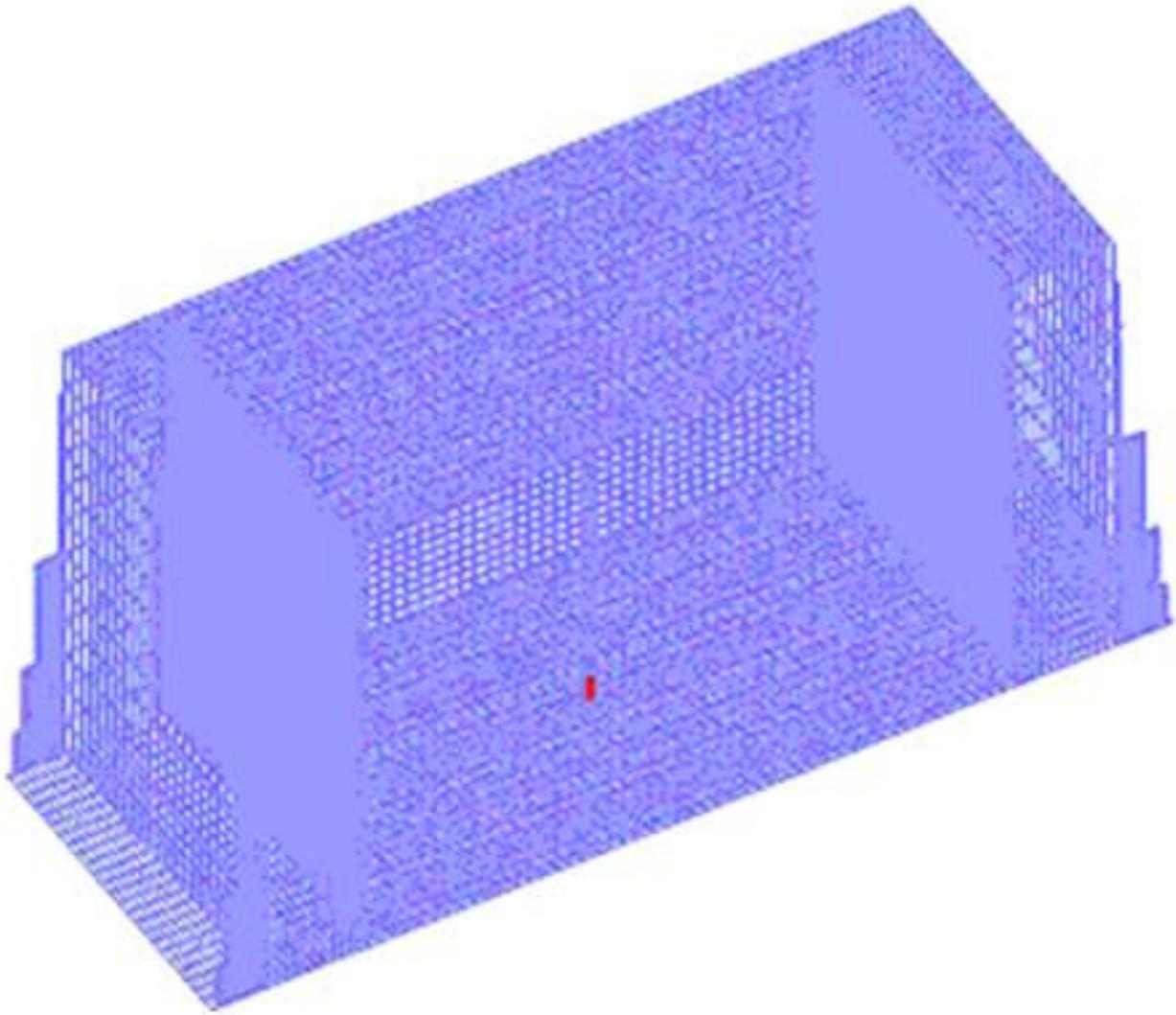


Рис. 2.9. Трехмерная тонкопроводная модель корпуса БА КА

2.8. Моделирование электромагнитной совместимости бортовой аппаратуры космических аппаратов

Рассмотрим общий случай моделирования ЭМС печатных плат БА КА (рис. 2.10). Процедура моделирования ЭМС корпуса блока БА является частным случаем процедуры, представленной на рис. 2.10, без квазистатического анализа.

На вход процедуры на рис. 2.10 подаются данные о геометрии, компонентах и материалах ПП блока БА КА,

которые либо вводятся пользователем с помощью средств графического или текстового ввода, либо импортируются из файлов формата САПР печатных плат. Введенная таким образом геометрическая математическая модель (ММ) ПП подается на вход двух блоков: блока электродинамического анализа и блока генерации ММ трасс, контактных площадок и переходных отверстий.

В случае электродинамического анализа пользователь вводит воздействие помехи, после чего выполняется тонкопроводная аппроксимация трасс ПП и электродинамический анализ полученной структуры. Результатом анализа являются значения $E(x, y, z)$ напряженности электрического поля в заданных точках ПП. Для этих точек в процессе ввода требований ЭМС пользователем задаются пороговые значения $E_{\max}(x, y, z)$, которые в блоке обработки результатов сравниваются с расчетными. Если $E(x, y, z) < E_{\max}(x, y, z)$ для всех заданных точек (x, y, z) , то анализируемая ПП удовлетворяет требованиям ЭМС по максимальным значениям напряженности электрического поля.

В случае квазистатического анализа выполняется генерация двумерных геометрических ММ отрезков трасс ПП и трехмерных геометрических ММ контактных площадок и переходных отверстий, а также подготовка схемы соединений и ММ компонентов.

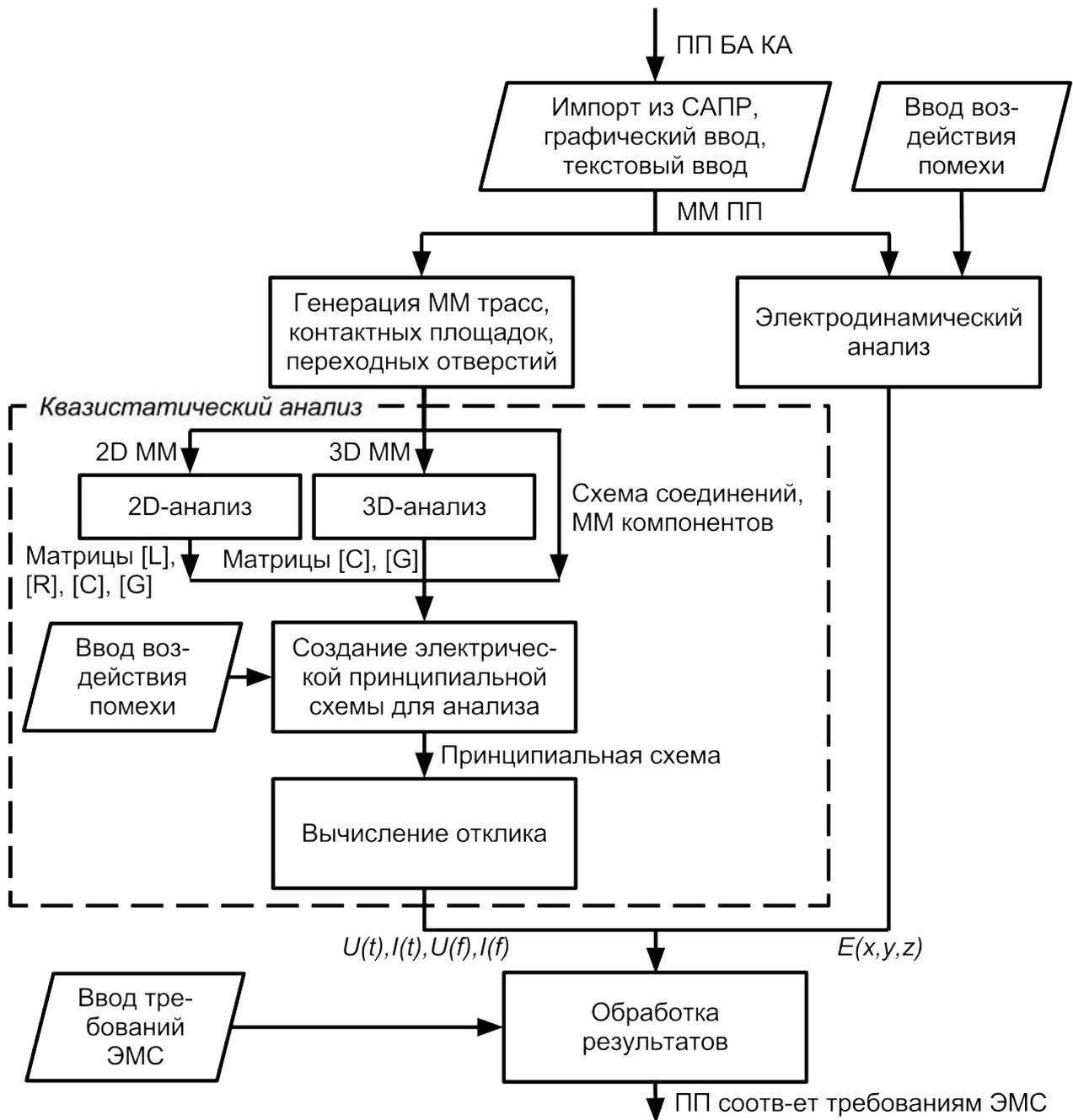


Рис. 2.10. Процедура моделирования ЭМС БА КА

Геометрические модели анализируются с целью получения матриц параметров (матрицы L, R, C, G при двумерном анализе и матрицы C, G при трехмерном). Затем с учетом введенного пользователем воздействия помехи (в виде источника помехового сигнала) создается электрическая принципиальная схема, для которой вычисляется

временной и частотный отклик. В блоке обработки результатов полученные значения $U(t)$, $I(t)$ и $U(f)$, $I(f)$ сигналов в узлах принципиальной схемы сравниваются с заданными пользователем пороговыми значениями. Если для всех временных и частотных точек значения $U(t)$, $I(t)$ и $U(f)$, $I(f)$ не превышают пороговые, то анализируемая ПП удовлетворяет требованиям ЭМС по максимальным значениям амплитуды сигналов.

Для программной реализации процедуры моделирования ЭМС БА КА – системы компьютерного моделирования и оптимизации ЭМС БА КА TALGAT [70] – выбран язык программирования C++, так как его компиляторы способны генерировать оптимизированный код, показывающий высокую производительность при решении сложных вычислительных задач, к которым относятся задачи моделирования ЭМС. Кроме этого, язык C++ стандартизирован, дополняется новыми возможностями и широко используется на разных компьютерных платформах и в разных операционных системах. Для языка C++ существует большое количество библиотек графических пользовательских интерфейсов, матричных операций, параллельных вычислений, трехмерной графики, в том числе свободно распространяемых. Язык C++ также используется в ряде коммерческих систем электродинамического моделирования, например в CST MWS.

3. Программная реализация системы компьютерного моделирования и оптимизации электромагнитной совместимости бортовой аппаратуры космических аппаратов

3.1. Структура системы TALGAT

Система компьютерного моделирования TALGAT [70–74] содержит развитые средства моделирования задач ЭМС БА КА и позволяет на их основе задавать и вычислять произвольные целевые функции.

В системе TALGAT действие пользователя над элементом графического интерфейса или ввод текстовой команды преобразуется клиентом ядра системы (исполняемый файл, запускаемый пользователем для начала работы с системой, QtClient.exe или BasiClient.exe) в текстовую команду и передается ядру системы (TLCORE.dll). Ядро системы распознает команду и вызывает ее обработчик, размещенный в соответствующем модуле (динамические библиотеки, экспортирующие специальные функции, с помощью которых ядро при загрузке модуля подключает содержащиеся в модуле математические модели и алгоритмы к системе).

К модулям утилит системы TALGAT относятся модуль команд общего назначения UTIL.tlm, модуль работы с матрицами MATRIX.tlm [87], модуль infixной записи выражений INFIX.tlm, модуль построения графиков GRAPH.tlm, модуль вычислений на графических ускорителях TLGPU.tlm и модуль импорта печатных плат TLPCB.tlm. К вычислительным модулям относятся модуль двухмерного квазистатического анализа

MOM2D.tlm [88, 89], модуль вычисления временного отклика RESPONSE.tlm, модуль трехмерного квазистатического анализа MOM3D.tlm и модуль электродинамического анализа трехмерных проводных структур MOMW.tlm. К модулям оптимизации относятся модуль эволюционных стратегий ES.tlm (также содержит реализацию алгоритма неявного фильтрования) и модуль генетических алгоритмов GA.tlm.

Структура системы TALGAT в части моделирования ЭМС БА КА показана на рис. 3.1.

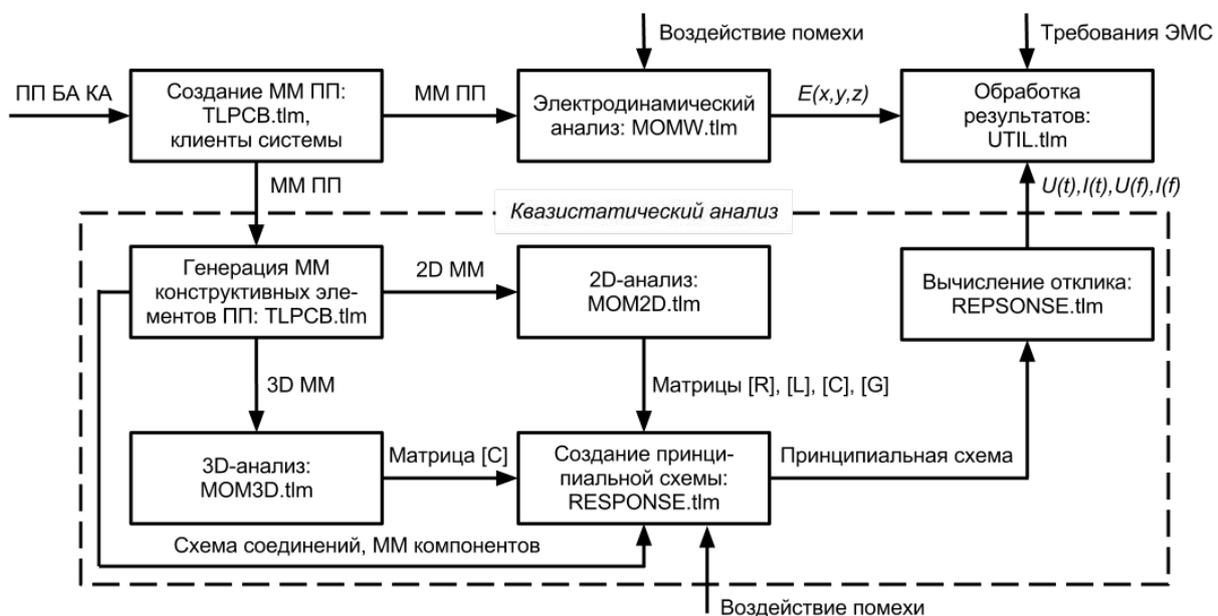


Рис. 3.1. Структура системы TALGAT в части моделирования ЭМС БА КА

В соответствии с рис. 3.1 система TALGAT состоит из программно реализованных модулей, объединенных в 8 блоков. Блок создания ММ печатной платы БА КА состоит из модуля TLPCB.tlm, кроме того, клиент системы QtClient.exe обеспечивает графический ввод, а клиент BasiClient.exe – текстовый. Блок электродинамического ана-

лиза реализован в модуле MOMW.tlm. Блок квазистатического анализа включает модули TLPCB.tlm, MOM2D.tlm, MOM3D.tlm и RESPONSE.tlm. Блок обработки результатов реализован средствами модуля UTIL.tlm.

Рассмотрим элементы структуры системы TALGAT более подробно.

3.2. Ядро системы TALGAT

В соответствии с поставленными задачами сформулированы требования к программной реализации.

1. Реализация различных методов для анализа структур проводников и диэлектриков, возможность анализа конкретной структуры несколькими методами и сравнения их результатов.
2. Открытость для включения новых моделей, анализирующих все более сложные структуры, а также относительно простые структуры, но с большей эффективностью.
3. Возможность синтеза и оптимизации структур.

Перечисленные требования предопределили максимальное использование достоинств объектно-ориентированного программирования [90, 91] и послужили отправной точкой для создания ядра системы TALGAT, которое является стержневым ее элементом.

Структура классов ядра системы TALGAT [77] представлена на рис. 3.2. Заметим, что классы CoreThread и Main относятся к клиентам системы, но приведены на рис. 3.2 для ясности.

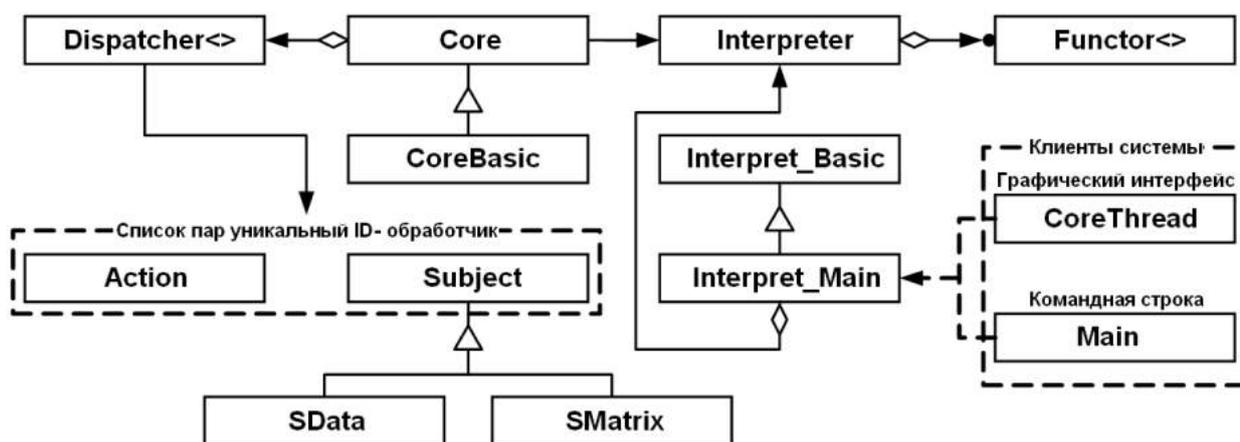


Рис. 3.2. Структура классов ядра системы TALGAT

На рис. 3.2 для обозначения наследования классов используется треугольник, направленный от подкласса к родительскому классу; отношение агрегирования обозначается стрелкой с ромбиком, при этом, если агрегируется несколько объектов, возле стрелки появляется заливный круг; отношение осведомленности (например, Dispatcher<> содержит информацию об уникальных идентификаторах всех подклассов Action и Subject) обозначается простой стрелкой; отношение «создает» (например, экземпляр класса Interpret_Main создается в методе класса Main клиента командной строки) обозначается пунктирной стрелкой.

Класс Interpreter реализует скриптовый язык системы TALGAT и предназначен для интерпретации потока ввода. Входным потоком может быть не только командная строка или файл, но и графическая среда, где действия пользователя над элементами управления конвертируются в поток текстовых команд. Важной особенностью скриптового языка системы TALGAT является возможность использовать в качестве параметра одной команды результат, возвращаемый другой командой, и возможность создания пользователем собственных команд на скриптовом языке.

Экземпляр класса `Interpreter` содержит хеш-таблицу с парами «строковый идентификатор команды – обработчик». Обработчик выполняет соответствующую команде операцию и может находиться в любом элементе системы TALGAT – в ядре, модуле или клиенте. В классе `Interpreter` применены паттерны `Facade`, `Command` [92] и обобщенные функторы [90].

Абстрактный класс `Interpret_Basic` является базовым для всех классов, содержащих обработчики команд. Производный от `Interpret_Basic` класс `Interpret_Main` агрегирует класс `Interpreter` и используется в главных процедурах системы (например, функция потока ядра в визуальном клиенте). `Interpret_Main` инициализируется входным потоком и регистрирует стандартные команды (загрузка модуля, работа с переменными, справка, завершение работы и другие реализованные в ядре системы команды). С помощью метода `Interpret` запускается обработка входного потока.

В основу функциональной части ядра системы TALGAT положены базовые классы `Action` (действие) и `Subject` (субъект). Когда на экземпляр класса `Subject` «действует» экземпляр класса `Action`, `Dispatcher` определяет реальный тип взаимодействующих объектов и вызывает заранее зарегистрированную функцию, которая и обрабатывает вызов. В классах `Action` и `Subject` применены паттерн `Visitor` и мультиметоды с постоянным временем выполнения.

Производные от класса `Subject` классы `SData` и `SMatrix` реализуют типы данных, доступные в скриптовом языке системы TALGAT. Экземпляр класса `SData` может содержать целое число, число с плавающей точкой, комплексное число, строку, экземпляр `SMatrix`, указатель на объект `Subject` (фактически, указатель на экземпляр любого

производного от Subject класса, реализованного в системе). Параметрами и возвращаемыми значениями обработчиков команд являются экземпляры класса SData, чем обеспечивается возможность использовать в качестве параметра одной команды результат, возвращаемый другой командой. Класс SMatrix реализует идиому Pimpl [91] для сокрытия реализации и предоставляет через единый интерфейс возможность работы с матрицами чисел с плавающей точкой и матрицами комплексных чисел.

Класс Core предназначен для объединения данных, относящихся к ядру системы в целом (двойной диспетчер функций с постоянным временем поиска Dispatcher, обработка исключений, сведения о версии ядра), и сокрытия системно-зависимого кода (динамическая загрузка модулей, полоса прогресса). Клиентами системы TALGAT реализуются классы, производные от класса Core. Конкретный тип производного класса выбирается в зависимости от потребностей клиента, в частности для клиентов командной строки заранее реализован класс Core_Basic, предоставляющий текстовую «полосу прогресса». В классе Core применен паттерн Singleton.

Таким образом, ядро системы TALGAT отвечает первым двум требованиям, изложенным выше: ядро позволяет использовать различные методы анализа одной структуры и сравнивать их результаты; с помощью механизма динамической загрузки исполняемых модулей можно увеличивать количество моделей анализа структур без перезапуска системы.

3.3. Язык скриптов

TALGAT_Script – встроенный в систему TALGAT интерпретируемый скриптовый язык, с помощью которого пользователь «общается» с системой. В качестве синтаксиса используется польская нотация. TALGAT_Script создан с учетом следующих требований, перечисленных в порядке их приоритета при разработке языка:

- предоставление пользователю как можно большего числа возможностей системы, доступных при программировании на C++;
- простота реализации;
- простота языка;
- быстрое действие.

Синтаксис. Скрипт на языке TALGAT_Script представляет собой последовательность символов – поток ввода (input stream). Не имеет никакого значения источник потока ввода. Это может быть файл на диске, буфер в памяти или текстовое поле визуального клиента системы.

В ядро системы встроен интерпретатор, распознающий во входном потоке команды (commands). Команда – это последовательность символов, не содержащая символов-разделителей (пробел, табуляция), символов «'», «"» и символа новой строки. Каждая команда во входном потоке должна начинаться с новой строки. Чтобы легко различать команды в больших скриптах, все команды системы пишутся заглавными буквами (например, «КОМАНДА»).

Команде соответствует обработчик (handler), выполняющий соответствующую операцию. Не имеет значения, где

находится обработчик – в ядре, модуле или клиенте системы. После распознавания команды интерпретатор вызывает соответствующий ей обработчик.

Команды могут иметь параметры (parameters), которые отделяются от команды и друг от друга символами-разделителями (пробел, табуляция). Например, «КОМАНДА это_параметр». Параметры могут иметь один из семи типов (types):

- пустой тип [NULL] или [null] – значение отсутствует (например, пользователь забыл его ввести);
- длинное целое long – 4-байтное знаковое целое число (например, «-44», «5»);
- двойной точности double – 8-байтное число с плавающей точкой (например, «0.256», «6.»²);
- строка string – последовательность символов, не содержащая символов-разделителей (пробел, табуляция, символ новой строки) и кавычек (например, «это_строка»);
- субъект subject – непрерывная область в памяти, содержащая данные определенного типа, в том числе конфигурации, структуры, матрицы (аналог объектов языка C++, при распечатке выводится в виде строки «[subject]»);
- комплексное число complex – два 8-байтных числа с плавающей точкой (например, «(0.344,-5.6)» – без пробела, точка также необходима, даже если после нее нет значащих цифр, например, «(0.,-5.)»);

² Для записи дробных чисел в системе вместо запятой используется точка, которая необходима, даже если после нее нет значащих цифр.

- матрица `matrix` – матрица из чисел типа `double` или `complex`, при этом матрицы из чисел типа `double` обозначаются в документации как тип `real_matrix`, матрицы из чисел типа `complex` – как тип `complex_matrix`.

Если команда имеет параметры, то обработчик запрашивает их значения у интерпретатора. Интерпретатор читает входной поток до конца текущей строки, распознает типы введенных параметров и передает их значения обработчику. Таким образом, не допускается наличие новых строк между командой и ее параметрами. Допускается использование в качестве параметра, тип которого должен быть, например `double`, параметр типа `long`, и наоборот. При этом выполняется автоматическое преобразование типов. Тем не менее, важно отличать: «4.» (четыре с плавающей точкой) имеет тип `double`, а «4» (просто четыре без точки) – тип `long` по аналогии с литералами C++.

После исполнения необходимых операций обработчик возвращает значение-результат (`value_result`), имеющее один из приведенных выше типов. Данное значение содержит результат проведенных операций и может использоваться как параметр другой команды. Таким образом, команды могут быть вложенными (`nested`), то есть вместо параметра команды пользователь может ввести имя другой команды, которая возвращает значение-результат того же типа, что и параметр первой команды. Например, «PLUS 2. PLUS 2. 2.» эквивалентно $2 + 2 + 2$.

Входной поток интерпретируется вплоть до получения команды «RETURN». Если используется клиент командной строки, то обнаружение такой команды закрывает клиент. Если используется графический клиент, то интерпретация

потока ввода (например, файла со скриптом) прекращается до поступления новых команд.

Стандартные команды. Стандартные команды (standard commands) – команды, которые реализованы в ядре системы и доступны сразу после запуска любого клиента системы. Рассмотрим стандартные команды более подробно.

Команда `INTERPRET` позволяет интерпретировать содержимое файла на диске с выводом результатов в другой файл. Для этого первым параметром команде передается имя входного файла, вторым – имя выходного. Кроме того, можно задавать в качестве одного или обоих файлов текущие устройства ввода-вывода³. Для этого существуют специальные имена файлов – `[STDIN]` для входного файла и `[STDOUT]` для выходного. Например, исполнение команды «`INTERPRET файл_на_диске [STDOUT]`» в командной панели графического клиента запустит интерпретацию файла «файл_на_диске» с выводом результатов в окно результатов командной панели. Интерпретация завершается при достижении команды «`RETURN`». Возвращаемым значением команды «`INTERPRET`» является параметр, используемый с командой «`RETURN`». Если таковой отсутствует, то возвращается 1 типа `long`.

Командой `INTERPRET` можно перенаправлять интерпретацию содержимого различных потоков ввода в различные потоки вывода. Потоками ввода и вывода могут являться командная панель и файлы. Например, ввод команды

³ Текущие устройства ввода-вывода – это потоки ввода-вывода, которые пользователь применяет для ввода команды, например, командная строка клиента командной строки или командная панель визуального клиента.

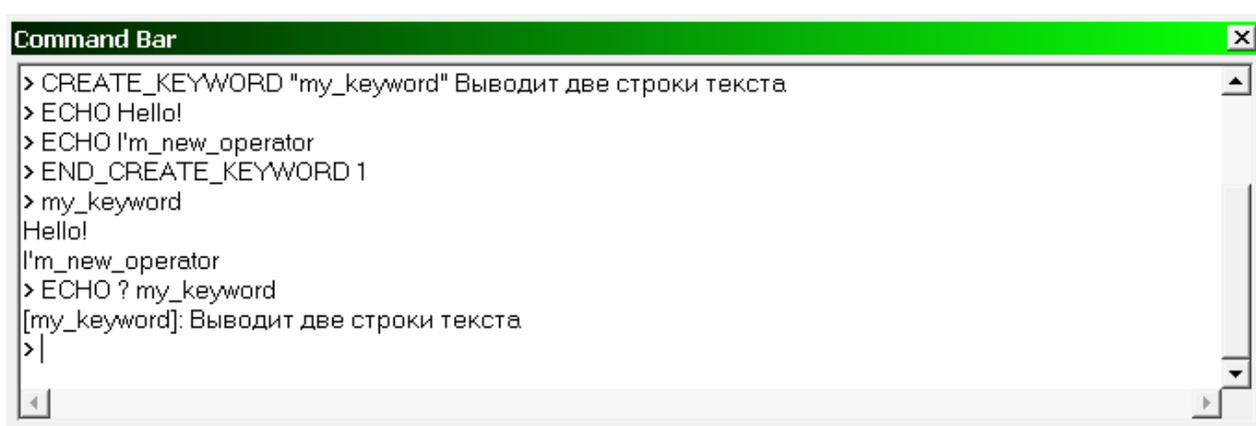
«INTERPRET [STDIN] out.txt» устанавливает в качестве потока ввода командную панель, а в качестве потока вывода – файл out.txt. Аналогично ввод команды «INTERPRET in.txt [STDOUT]» интерпретирует файл in.txt в стандартный поток вывода (командную панель).

Команда INCLUDE. позволяет загружать модули системы. Первый параметр – название модуля (обычно имя динамической библиотеки, в которой реализован модуль, без расширения). Второй необязательный параметр – имя подмодуля. Если имя подмодуля не задано, оно считается тем же, что и имя модуля. В результате пользователь получает доступ к командам, реализованным в данном модуле.

Команды для создания динамических команд. Команда `CREATE_KEYWORD` позволяет создавать динамические команды (dynamic commands). Первый параметр – имя новой команды, заключенное в кавычки, второй – строка, принимаемая за результат команды «HELP имя_динамической_команды» (комментарий к создаваемой команде, например краткая инструкция по ее использованию). При этом интерпретатор переходит в специальный режим и сохраняет поток ввода как строку, пока не будет введена команда `END_CREATE_KEYWORD`, которая возвращает интерпретатор в обычный режим. Теперь пользователю доступна новая команда, результат которой равен введенному ранее параметру команды `END_CREATE_KEYWORD`.

Пример использования команды `CREATE_KEYWORD` приведен на рис. 3.3. Здесь пользователь создает новую

команду с именем «my_keyword», которая будет выводить две строки: «Hello!» и «I'm_new_operator», посредством последовательного выполнения двух команд ECHO. Поэтому пользователь указывает в комментарии к создаваемой команде, что она «выводит две строки текста». Затем пользователь вводит две команды ECHO, которые и будут выполняться при каждом вызове создаваемой новой команды. Создание новой команды завершается вводом команды «END_CREATE_KEYWORD 1». Затем пользователь путем ввода имени «my_keyword» вызывает новую команду, которая, как и предполагалось, выводит две строки с вышеуказанным текстом. При этом результатом ее выполнения будет число 1, так как именно оно было указано в качестве параметра команды END_CREATE_KEYWORD во время создания новой команды. Кроме того, пользователь проверяет, сохранился ли в памяти его комментарий к созданной им команде «my_keyword» путем ввода команды «ECHO ? my_keyword». Действительно, команда ECHO выводит именно тот комментарий, который был ранее задан пользователем для команды «my_keyword».



```
Command Bar
> CREATE_KEYWORD "my_keyword" Выводит две строки текста
> ECHO Hello!
> ECHO I'm_new_operator
> END_CREATE_KEYWORD 1
> my_keyword
Hello!
I'm_new_operator
> ECHO ? my_keyword
[my_keyword]: Выводит две строки текста
> |
```

Рис. 3.3. Пример использования команды CREATE_KEYWORD

Пользователь может переопределять (override), то есть создавать заново, ранее созданные динамические команды, однако это запрещено для статических команд (static commands), под которыми понимаются команды, реализуемые системой, в частности стандартные команды системы, команды из модулей системы. Если при загрузке нового модуля системы выяснится, что пользователь создал команду с именем, совпадающим с именем команды из модуля, то динамическая команда будет удалена (без сообщения об этом пользователю).

Создание вложенных динамических команд не поддерживается. Если после вызова команды `CREATE_KEYWORD` и перехода интерпретатора в специальный режим команда `CREATE_KEYWORD` используется еще раз (до считывания команды `END_CREATE_KEYWORD`), то она просто сохраняется как строка и при последующем вызове созданной динамической команды происходит ошибка. Если необходимо использовать внутри динамической команды с именем «my_keyword1» другую динамическую команду «my_keyword2», следует создать динамическую команду «my_keyword2» заранее, перед созданием «my_keyword1».

Команды SET и GET: работа с переменными. Команда `SET` присваивает значение переменной, первый параметр – имя переменной, заключенное в кавычки, например «"имя_переменной"», второй параметр – значение переменной. Команда `SET` устанавливает значение переменной, указанной в качестве первого параметра, в значение второго параметра. Если переменная с таким именем не существует, то она будет создана. Имя переменной нужно указывать

в кавычках. Например, «SET "var" 55.2» установит значение переменной var равным 55.2 (листинг 3.1).

Команда GET позволяет получать значение переменной, первый параметр – имя переменной, заключенное в кавычки, например «"имя_переменной"». Если переменной с таким именем не существует, генерируется исключение.

Листинг 3.1 Команды SET и GET

```
SET "var" 55.2 // записывает значение переменной "var"  
ECHO GET "var" // выводит 55.2
```

Помимо этого, команда SET создает для удобства пользователя новую динамическую команду с таким же именем, как и у переменной. Эта команда при вызове «подставляет» вместо себя значение одноименной переменной, например «ECHO var». Так, в листинге 3.2 создается команда var, которая работает аналогичным образом.

Листинг 3.2 Команда var

```
CREATE_KEYWORD "var" // создает команду var  
  RETURN GET "var"  
END_CREATE_KEYWORD  
ECHO var // выводит 55.2
```

Обработка ошибок. В случае ввода команд с ошибками интерпретатор генерирует исключение C++, которое перехватывается и обрабатывается клиентом системы. Клиент командной строки выводит сообщение об ошибке в текстовом виде, в том числе описание ошибки и строку скрипта, на которой произошла ошибка. Если ошибка возникла

внутри динамической команды или команды цикла, распечатывается также стек вызовов (по убыванию уровня вложенности вызова команды). Пример для ошибки в команде `add_vswr`, которая вызывается в цепочке вложенных команд `FOR_RANGE – CallQF1 – qf – qf_unconstrained – FOR`, приведен в листинге 3.3.

Листинг 3.3 Распечатанный стек вызовов при ошибке в команде `add_vswr`

```
Exception occurred on line 'FOR "f"
10.e+6 30.e+6 10.e+6 add_vswr'.
Exception occurred on line '
THEN SET "retval" qf_unconstrained'.
Exception occurred on line 'NULL qf'.
Exception occurred on line 'SET "m1"
FOR_RANGE "var" 5 18 1 CallQF1'.
```

Графический клиент при обработке ошибки создает окно исключения (рис. 3.4) с информацией об ошибке.

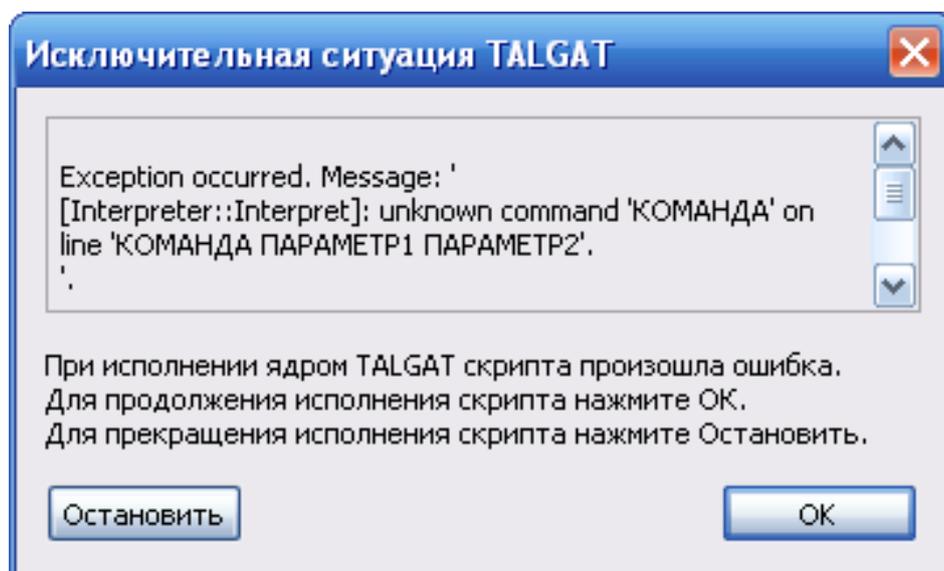


Рис. 3.4. Окно с информацией об исключении

Окно исключительной ситуации содержит текстовое описание ошибки; текст строки, которая вызвала ошибку; кнопку ОК для продолжения работы; кнопку «Остановить», которая позволяет предотвратить повторное исполнение команды, если она включена в цикл.

3.4. Динамические модули

Рассмотрим реализацию динамических модулей системы TALGAT на примере модуля матричных операций MATRIX. Модуль MATRIX предоставляет пользователю команды для работы со встроенными типами матриц системы TALGAT. Процедура загрузки модуля MATRIX представлена на рис. 3.5.

Функция `ModuleOpen_MATRIX()` создает экземпляр класса `Interpret_MATRIX`, производного от класса `Interpret_Basic`, конструктор которого выполняет регистрацию команд модуля MATRIX. Для этого используется метод `CreateKeyword()` класса `Interpreter`, принимающий в качестве параметров имя команды и ссылку на обработчик команды в виде экземпляра шаблона `Functor<>`. Шаблон `Functor<>` может быть инстанцирован ссылкой на простую функцию или метод класса. В любом из этих случаев диаграмма взаимодействий во время обработки команды пользователя схематично представлена на рис. 3.6.

На рис. 3.6 и далее для обозначения времени жизни объекта используется сплошная вертикальная линия; время, когда объект не существует, обозначено вертикальной пунктирной линией; время, в течение которого объект активен, обозначается вертикальным прямоугольником; запрос к другому объекту обозначается горизонтальной линией

со стрелкой, на которой указывается содержание запроса; запрос на создание объекта обозначается пунктирной линией со стрелкой.

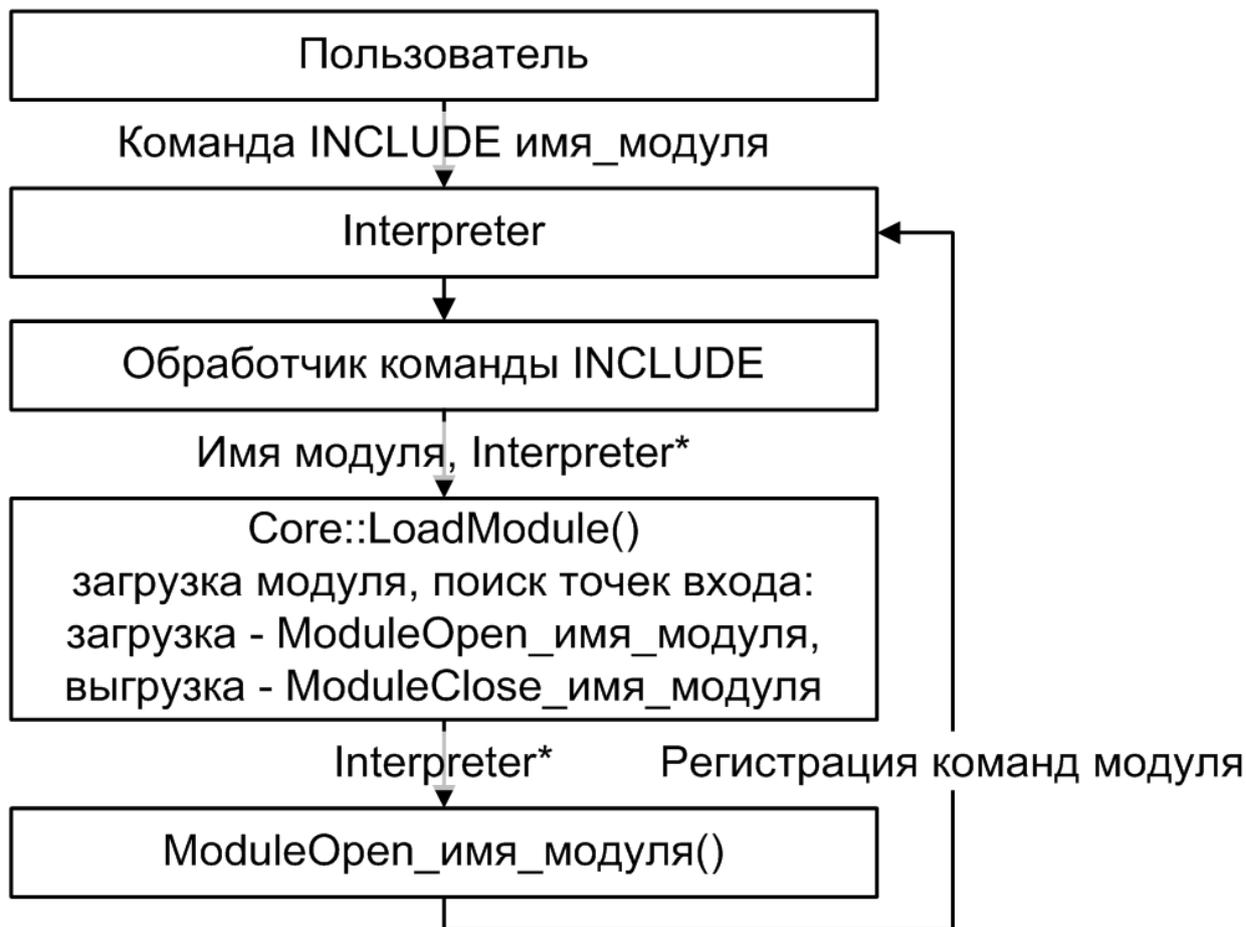


Рис. 3.5. Процедура загрузки динамического модуля

Помимо регистрации команд, функция `ModuleOpen_MATRIX()` регистрирует в `Dispatcher` пары производных от классов `Action` и `Subject`, в частности регистрирует пару классов `AEqu` и `SComplexMatrix` и сопоставляет данной паре обработчик `Equ_ComplexMatrix`. Производный от класса `Action` класс `AEqu` реализован в модуле `UTIL` и предоставляет пользователю возможность использовать команду `EQU` для сравнения любых типов данных системы, в том числе реализованных

в динамически загружаемых модулях. Диаграмма взаимодействий при сравнении командой EQU двух комплексных матриц $m1$ и $m2$ (объектов класса `SComplexMatrix`) представлена на рис. 3.7. В начале сравнения типы сравниваемых объектов неизвестны.

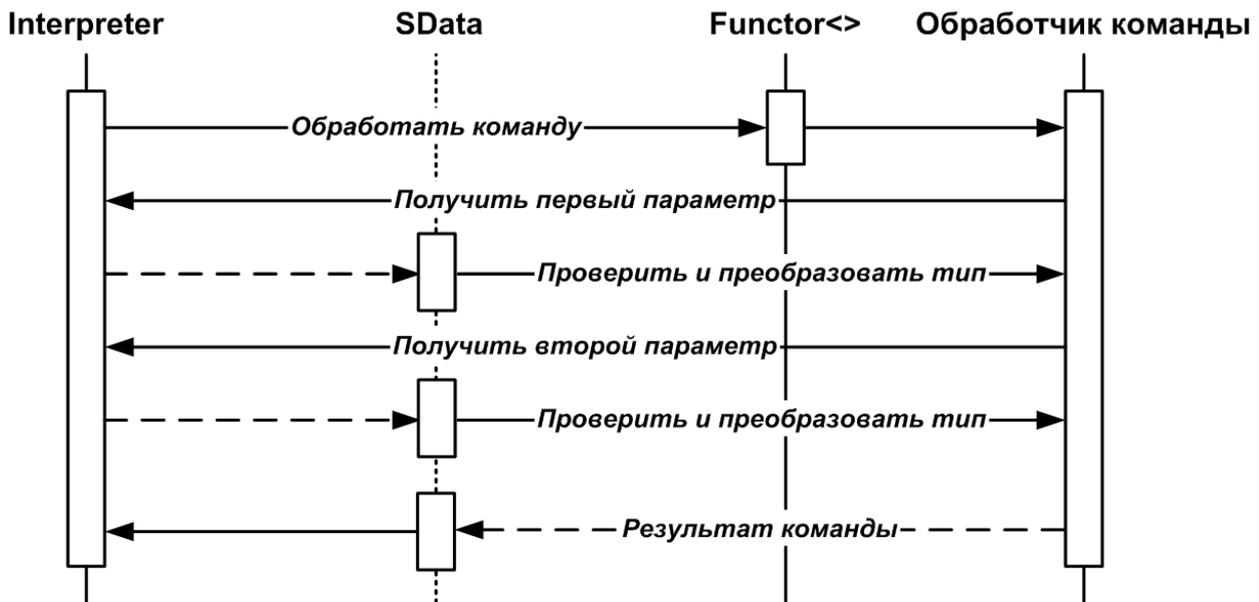


Рис. 3.6. Диаграмма взаимодействий во время обработки команды пользователя (`Functor<>` инстанцирован ссылкой на функцию или метод класса)

Из рис. 3.7 видно, что двойной диспетчер функторов с постоянным временем поиска `Dispatcher` позволяет реализовать «перегруженные» команды, обработчики которых выбираются на основании типов параметров. Такие команды полностью реализуются в динамических модулях системы TALGAT, поэтому создание новых «перегруженных» команд не требует модификации ядра системы. Обработчики «перегруженных» команд для разных сочетаний параметров могут располагаться в разных модулях системы. Таким образом, предоставление пользователю возможности сравнивать командой EQU объекты классов из любого нового

модуля системы TALGAT не требует модификации каких-либо других модулей системы.

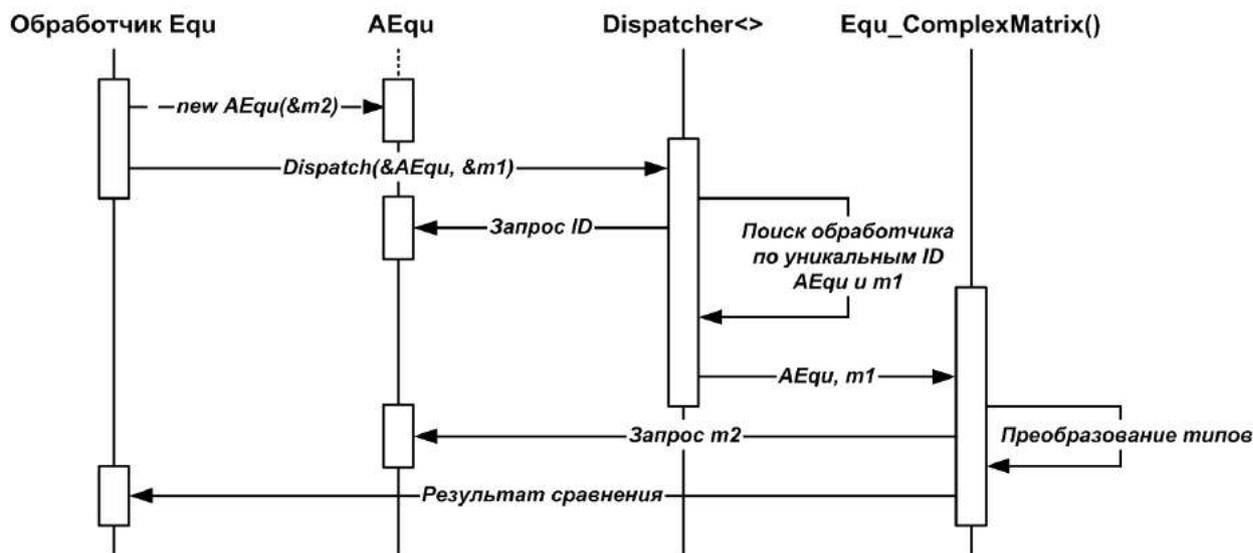


Рис. 3.7. Диаграмма взаимодействий при сравнении двух комплексных матриц m1 и m2 (в начале сравнения типы объектов неизвестны)

3.5. Клиенты системы TALGAT

Простейшим клиентом системы TALGAT является клиент командной строки, обеспечивающий работу пользователя с системой в диалоговом режиме через командную строку. Функция `main()` клиента командной строки инициализирует экземпляр класса `CoreBasic` и экземпляр класса `Interpret_Main` стандартными потоками `cin` и `cout` и запускает интерпретацию входного потока. Обработка ошибок заключается в использовании конструкции `try-catch` для перехвата всех исключений (исключений системы TALGAT, библиотеки STL, неизвестных исключений) и вывода информации об ошибке.

Графический клиент системы TALGAT [93, 94] обеспечивает работу пользователя через графический интерфейс пользователя. Графический клиент является

многопоточным приложением, в котором ядро системы TALGAT функционирует в отдельном потоке, что позволяет обрабатывать промежуточные результаты вычислений (например, диаграммы направленности синтезированных в процессе оптимизации антенн) во время работы потока ядра.

Функция потока ядра графического клиента аналогична функции `main()` клиента командной строки. В качестве потоков ввода-вывода используются экземпляры классов `istream` и `ostream`, инициализированные буферами `CoreInBuf` и `CoreOutBuf`. Запись в `CoreInBuf` входного скрипта запускает поток ядра. По мере поступления в `CoreOutBuf` результатов выполнения скрипта информация выводится в окно графического клиента. Вместо класса `CoreBasic` используется реализованный в графическом клиенте класс `VisualCore`. Класс `VisualCore` предоставляет возможность модулям системы производить графический вывод в окно графического клиента и обрабатывать действия пользователя (мышь, клавиатура и т.д.). Для графического вывода модули системы TALGAT регистрируют специальные команды, обработчики которых выполняют динамическое приведение `dynamic_cast` объекта ядра системы к типу `VisualCore`. Если приведение окончилось успешно, то модуль загружен графическим клиентом и выполняется графический вывод. В ином случае генерируется исключение.

Обработка ошибок реализована следующим образом. Если пользователем включен «тихий» режим обработки ошибок, информация о сгенерированных во время выполнения скрипта исключениях выводится в окно вывода и выполнение скрипта продолжается. Если «тихий» режим отключен,

то при генерации исключения выводится окно с информацией об ошибке и кнопками, позволяющими продолжить или остановить дальнейшее выполнение скрипта. Информация об исключении в этом случае также выводится в окно вывода.

Схема алгоритма графического клиента системы TALGAT представлена на рис. 3.8 и 3.9.

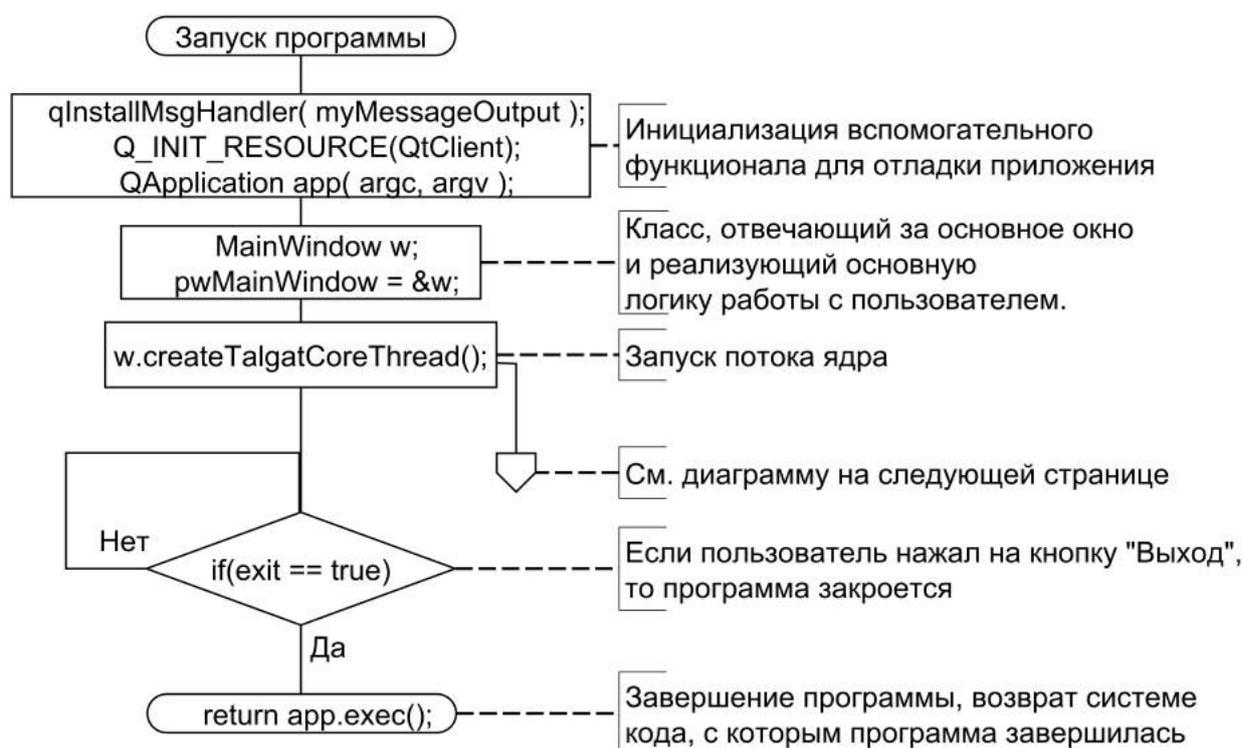


Рис. 3.8. Алгоритм графического клиента: графический поток

После запуска главной программы системы TALGAT (см. рис. 3.8) происходит инициализация вспомогательного функционала для контроля корректности функционирования приложения и создание объекта `w` класса `MainWindow`. Класс `MainWindow` создает главное окно и реализует основную логику работы с оператором. После главного окна создается поток ядра (см. рис. 3.9), который выполняется в фоновом режиме. Затем проверяется, нажал ли оператор

на кнопку «Выход». Если кнопка «Выход» была нажата, то главная программа закрывается, если нет, то продолжит работу.

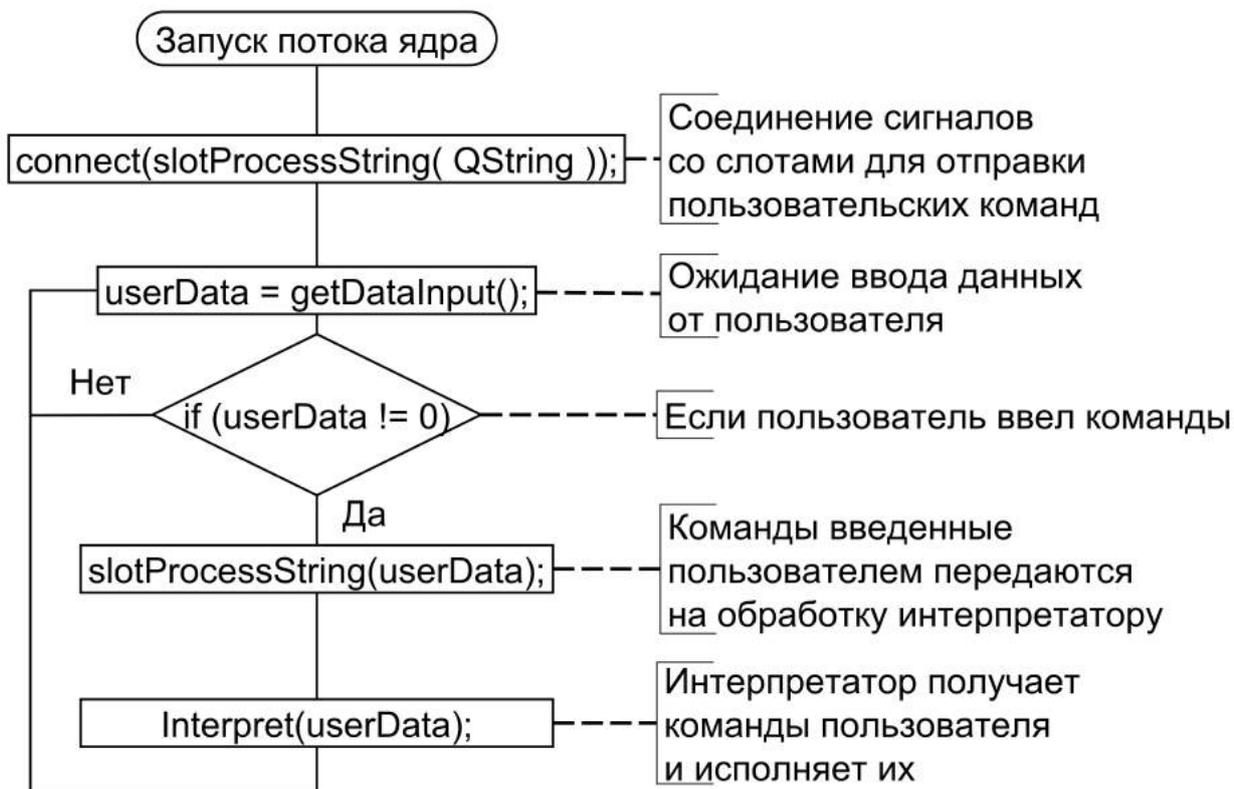


Рис. 3.9. Алгоритм графического клиента: поток ядра

После запуска графическим потоком потока ядра (см. рис. 3.9), выполняется инициализация соединений сигналов со слотами для отправки пользовательских команд. Далее поток ядра ожидает ввода данных от пользователя и проверяет в цикле, ввел ли пользователь новые команды. После получения новые команды передаются на обработку интерпретатору, который их исполняет.

3.6. Тестирование системы TALGAT

Рассмотрим процедуру математического моделирования ЭМС унифицированных электронных модулей (УЭМ) БА КА (рис. 3.10).

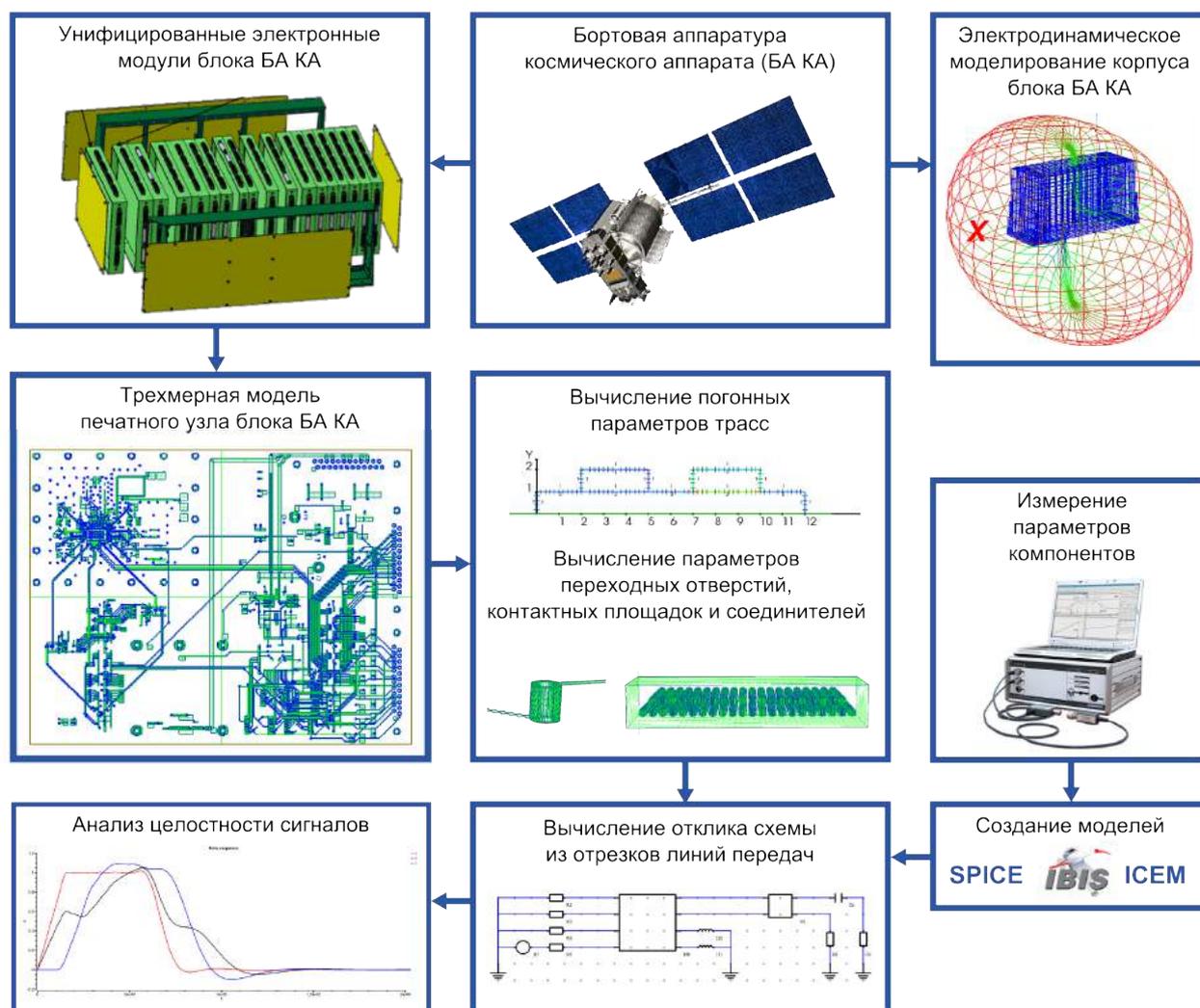


Рис. 3.10. Математическое моделирование ЭМС УЭМ БА КА в системе TALGAT

Как видно на рис. 3.10, при моделировании ЭМС БА КА отдельно выполняется электродинамическое моделирование корпуса блока БА. Для находящихся внутри корпуса УЭМ осуществляется импорт печатного узла блока БА из формата САПР для проектирования печатных плат

(Altium Designer) [95] и построение трехмерной модели этого узла. На следующем шаге производится вычисление погонных параметров трасс и параметров переходных отверстий, контактных площадок и соединителей, после чего выполняется расчет отклика схемы из отрезков линий передачи. На этом шаге используются модели SPICE, IBIS и ICEM, созданные на основе измеренных параметров сосредоточенных компонентов (микросхем, транзисторов и т.д.). Последним этапом имитационного моделирования является анализ целостности сигналов рассматриваемого печатного узла.

Математическое моделирование ЭМС БА КА заключается в использовании программной реализации системы TALGAT для моделирования ряда элементов печатного узла блока БА КА: трасс [96], переходных отверстий, контактных площадок и т.д. При этом применяются разные виды анализа (квазистатический и электродинамический), выполняется расчет временного и частотного откликов линий передачи, что ставит качество результатов оптимизации в зависимость от точности имитационного моделирования.

Сравнение результатов натурального и вычислительного экспериментов на примере связанных линий печатной платы. Сравнение результатов натурального и вычислительного экспериментов выполнялось на примере связанных линий, расположенных на печатной плате 768.2750-800 УЭМ БА КА разработки ОАО «ИСС» (рис. 3.11).

Экспериментальная установка состояла из вычислительного комбинированного осциллографа С9-11, кабелей, аттенюатора 20 дБ, переходов типа СРГ-50. На плате было выбрано два фрагмента: фрагмент 1 – пара связанных ли-

ний на стороне А; фрагмент 2 – несколько связанных линий на правом краю стороны Б. При проведении эксперимента на один из проводников печатной платы подавался воздействующий сигнал в двух вариантах: проводник-металлическое основание; проводник-проводник (дифференциальное воздействие).

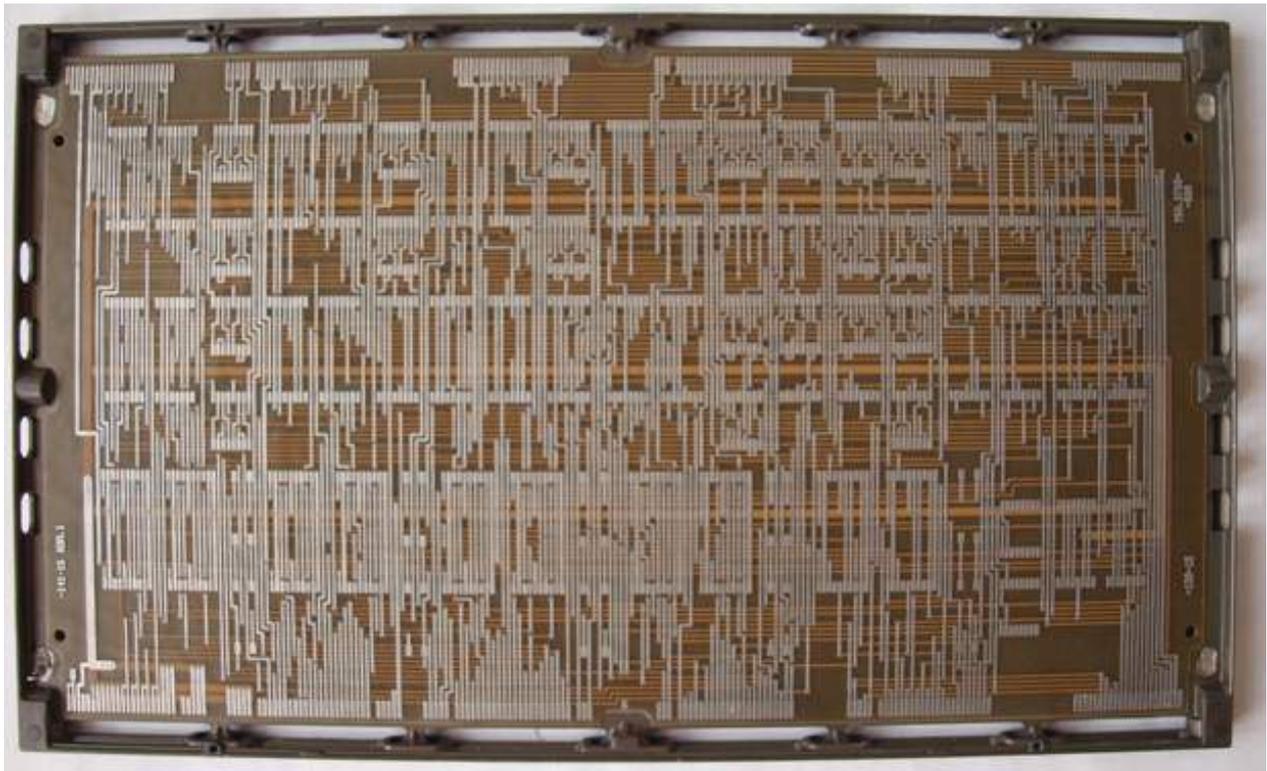


Рис. 3.11. Печатная плата УЭМ БА КА разработки ОАО «ИСС»

Пара связанных линий. Рассматриваемые линии на стороне А печатной платы изображены на рис. 3.12. Стрелками на печатной плате показаны контакты на концах пары связанных линий, к которым подсоединялась экспериментальная установка.

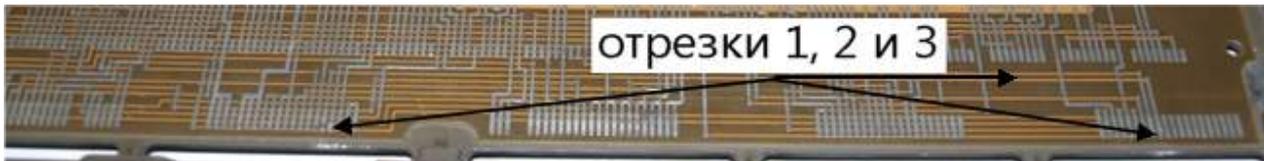


Рис. 3.12. Пара связанных линий на стороне А печатной платы

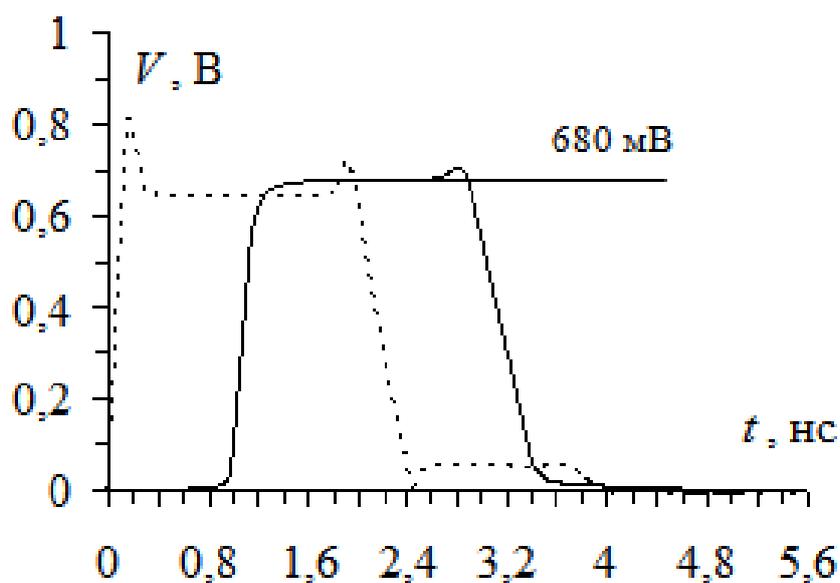
Моделирование выполнялось для измеренных значений относительной диэлектрической проницаемости материалов платы: 2,65 для СТФ и 3,96 для СТП. Параметры сигнала: амплитуда 1,4 В; $t_r = 160$ пс; $t_f = 520$ пс; $t_d = 1,75$ нс. Временной отклик вычислялся для трех последовательно соединенных отрезков линии передачи: отрезок 1 – ППЛ, отрезок 2 – ОПЛ и отрезок 3 – ППЛ. Параметры ППЛ: $l = 1,25$ см; $L_{11} = L_{22} = 387,38$ нГн/м, $L_{12} = L_{21} = 73,97$ нГн/м; $C_{11} = C_{22} = 63,16$ пФ/м, $C_{12} = C_{21} = -7,301$ пФ/м; значения элементов матриц R и G , вычисленных на частоте 1 МГц: $R_{11} = R_{22} = 0,639$ Ом/м, $R_{12} = R_{21} = 0,0167$ Ом/м (LINPAR), $G_{11} = G_{22} = 5,437$ мкСм/м, $G_{12} = G_{21} = -0,253$ мкСм/м. Параметры ОПЛ: $l = 13,8$ см; $L_{11} = L_{22} = 255,7$ нГн/м, $L_{12} = L_{21} = 22,031$ нГн/м; $C_{11} = C_{22} = 147,826$ пФ/м, $C_{12} = C_{21} = -10,389$ пФ/м; значения элементов матриц R и G , вычисленных на частоте 1 МГц: $R_{11} = R_{22} = 0,639$ Ом/м, $R_{12} = R_{21} = 0,0167$ Ом/м (LINPAR), $G_{11} = G_{22} = 21,166$ мкСм/м, $G_{12} = G_{21} = -1,051$ мкСм/м.

Формы сигналов в конце активной линии, полученные в системе TALGAT и в эксперименте, представлены на рис. 3.13, а и 3.13, б соответственно. Результаты согласуются. Амплитуда сигнала в конце активной линии, полученная в системе TALGAT, равна 680 мВ, а в эксперименте – 720 мВ. Различие между результатами натурального и вы-

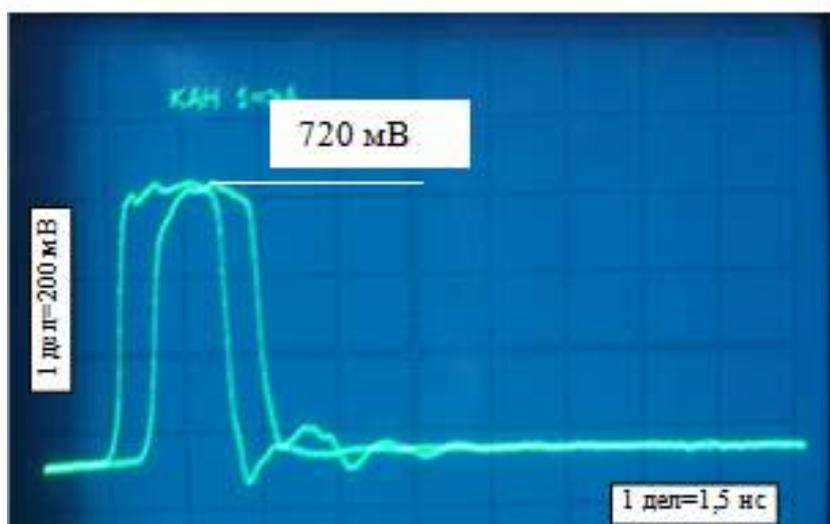
числительного экспериментов здесь и далее вычисляется по формуле

$$\Delta U = \frac{|U_{\text{Exp}} - U_{\text{TALGAT}}|}{U_{\text{Exp}}} \cdot 100 \%, \quad (3.1)$$

где U_{Exp} – амплитуда сигнала в натурном эксперименте;
 U_{TALGAT} – амплитуда сигнала в системе TALGAT, ее значение составляет 5,6 %.



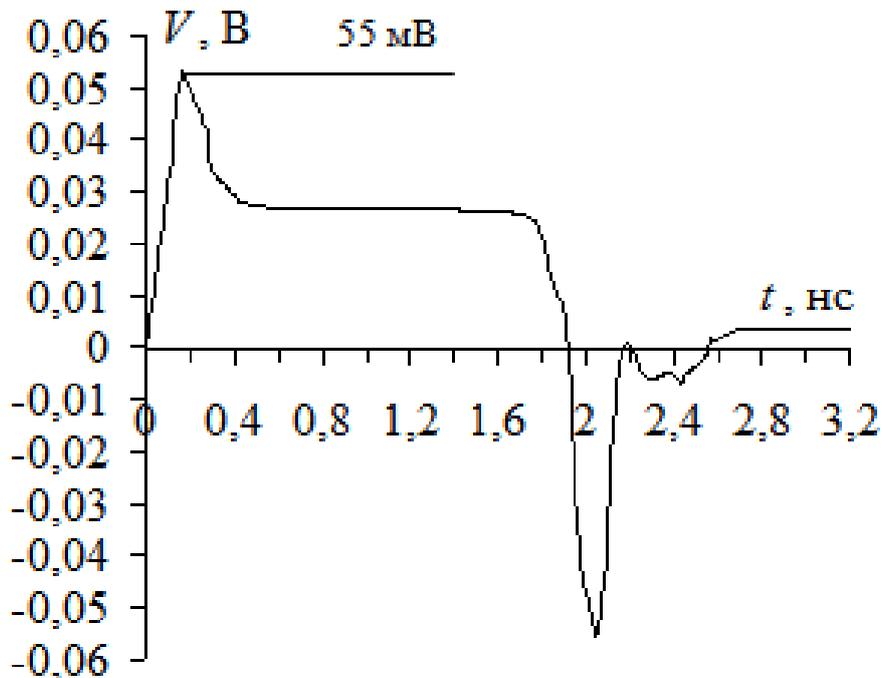
a



б

Рис. 3.13. Формы сигналов для связанной линии (см. рис. 3.12), полученные в системе TALGAT (*a*) и в натурном эксперименте (*б*)

Форма сигнала в начале пассивной линии показана на рис. 3.14. Амплитуда сигнала в начале пассивной линии, полученная в системе TALGAT (рис. 3.14, а), равна 55 мВ, а в эксперименте (рис. 3.14, б) – 64,5 мВ. Различие составляет 14 %.



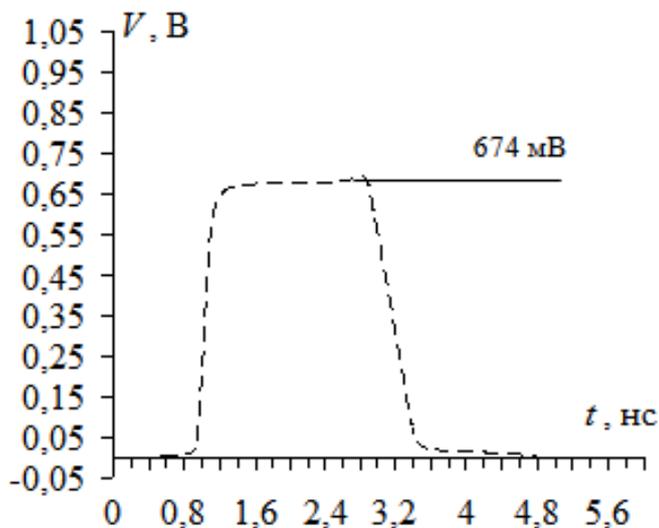
а



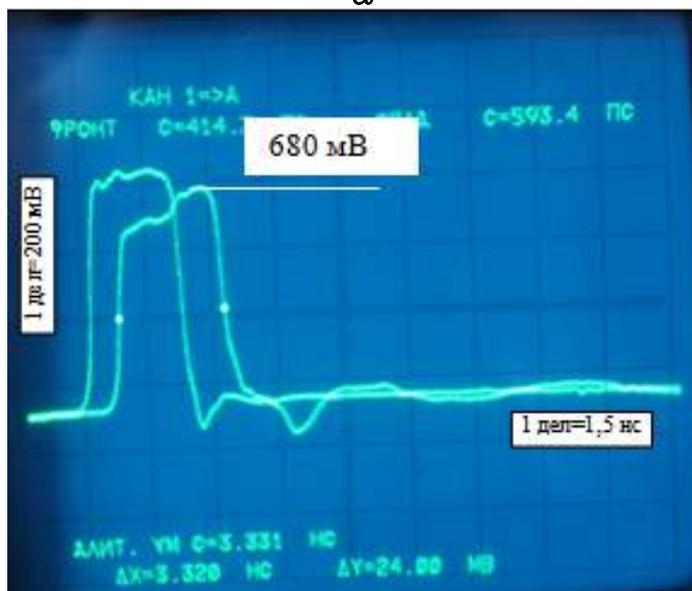
б

Рис. 3.14. Форма перекрестной помехи в начале пассивного проводника для связанной линии (см. рис. 3.12), полученная в системе TALGAT (а) и в натурном эксперименте (б)

Формы сигналов в конце активной линии для дифференциального воздействия показаны на рис. 3.15. Амплитуда сигнала в конце активной линии для дифференциального воздействия, полученная в системе TALGAT (см. рис. 3.15, а), равна 674 мВ, а в эксперименте (рис. 3.15, б) – 680 мВ. Различие составляет 0,9 %.



а



б

Рис. 3.15. Формы сигналов для связанной линии (см. рис. 3.12), полученные в программном комплексе TALGAT (а) и в натурном эксперименте (б)

Несколько связанных линий. Исследуемые связанные линии на правом краю стороны В печатной платы показаны на рис. 3.16. Временной отклик вычислялся для двух последовательно соединенных отрезков линии передачи (показаны на рис. 3.16 стрелками) с длинами $l_1 = 1,075$ см и $l_2 = 4,191$ см. Поперечные сечения структур, построенные в системе TALGAT, представлены на рис. 3.17.

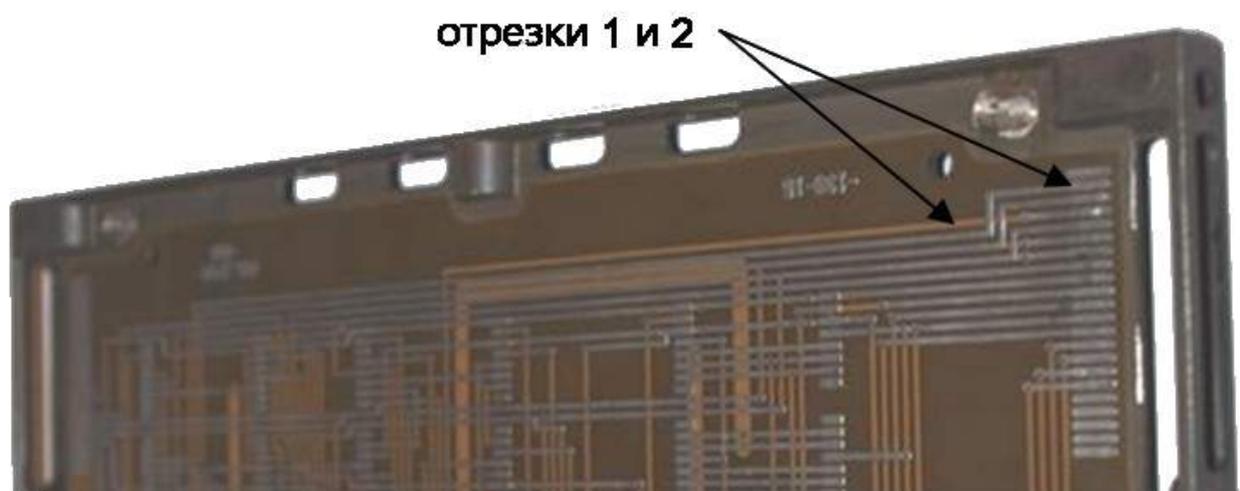


Рис. 3.16. Несколько связанных линий на правом краю стороны В печатной платы

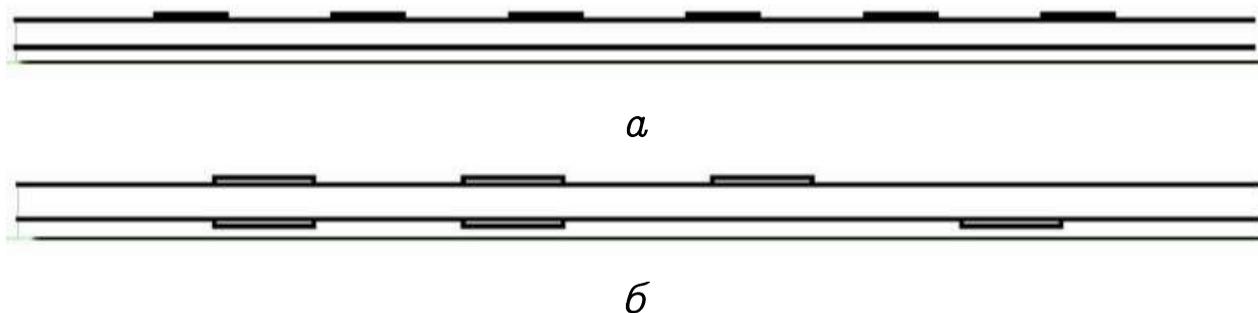
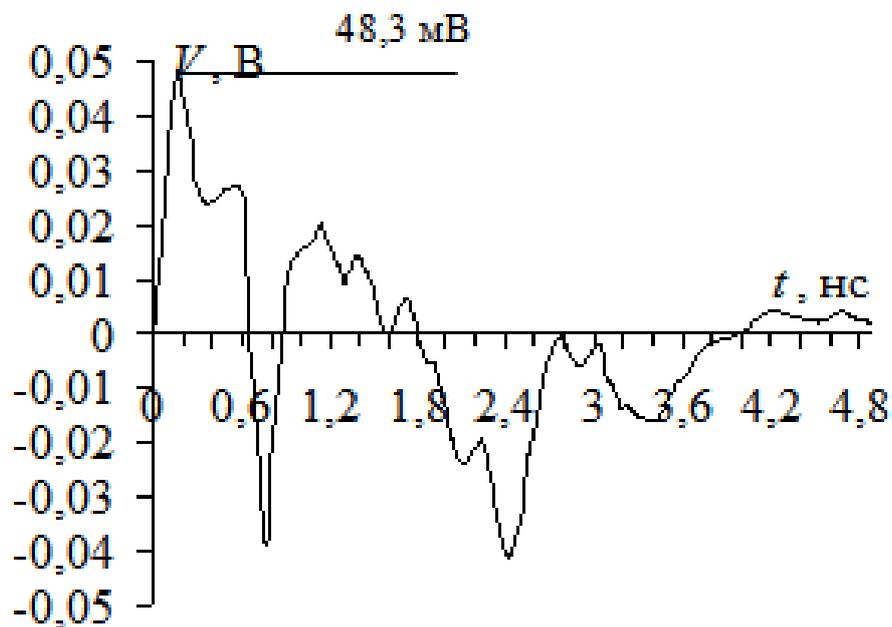


Рис. 3.17. Моделирование ППЛ (а) и ПОПЛ (б) в системе TALGAT

Форма сигнала на пассивной линии (вторая сверху) при воздействии на первую линию сверху показана на рис. 3.18. Амплитуда сигнала в конце пассивной линии, полученная

в системе TALGAT (рис. 3.18, *а*), равна 48,3 мВ, а в эксперименте (рис. 3.18, *б*) – 52 мВ. Различие составляет 7,1 %.



а

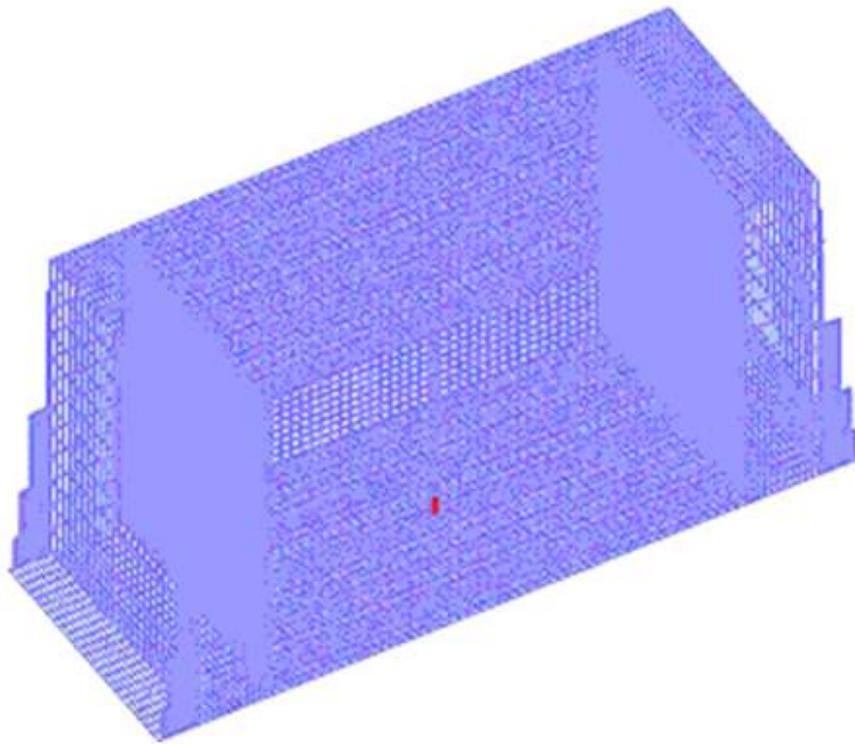


б

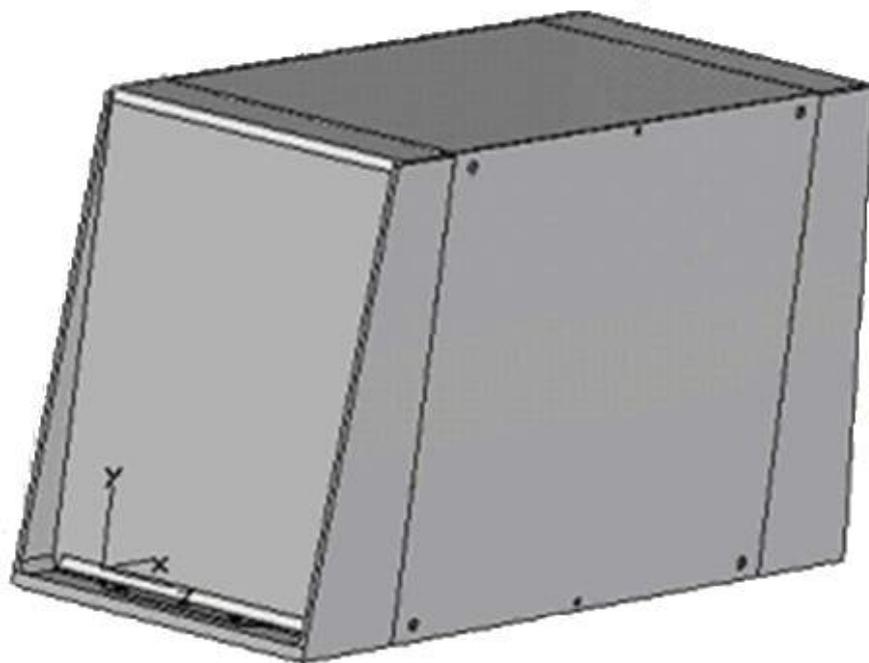
Рис. 3.18. Формы перекрестных помех на ближнем конце пассивной линии, полученные в системе TALGAT (*а*) и в натурном эксперименте (*б*)

Анализ полученных результатов. Таким образом, реализованные в системе TALGAT вычислительные модули позволяют выполнять имитационное моделирование печатных узлов БА КА с удовлетворительной точностью: форма вычисленных сигналов согласуется с результатами натурального эксперимента, различие в амплитуде сигналов составляет менее 15 %. Причина различий заключается в том, что при моделировании в системе TALGAT учитывались потери в линии передачи, но не учитывались неоднородности.

Сравнение результатов электродинамического анализа в системах TALGAT и CST MWS на примере корпуса бортовой аппаратуры. Для тестирования модуля электродинамического анализа MOMW в системе TALGAT выполнено моделирование корпуса блока БА (рис. 3.19, а). При моделировании поверхностей корпуса использовалась тонкопроводная аппроксимация. Для сравнения моделирование корпуса выполнено в системе CST MWS (рис. 3.19, б) путем аппроксимации объема прямоугольными ячейками. В диапазоне частот 0,1–2 ГГц вычислены частотные зависимости входного импеданса несимметричного вибратора длиной 12 см, расположенного в центре корпуса БА. Значения относительных разностей резонансных частот модуля входного импеданса, вычисленные в системах TALGAT (f_{TL}) и CST MWS (f_{CST}), приведены в табл. 3.1.



a



б

Рис. 3.19. Корпус блока БА в системах TALGAT (*a*) и CST MWS (*б*)

Таблица 3.1

Значения относительных разностей
резонансных частот модуля входного импеданса

f_{TL} , МГц	f_{CST} , МГц	$ f_{CST} - f_{TL} / f_{CST} \cdot 100\%$
500	560	12
1415	1460	3,1
1780	1830	2,8

Из данных табл. 3.1 видно, что различия резонансных частот во всем диапазоне составляют не более 12 %. Значения f_{CST} считаются в данной работе эталонными в связи с тем, что система CST MWS является лидером рынка ПО для электродинамического моделирования.

Таким образом, система TALGAT выполняет электродинамическое моделирование корпуса блока БА КА в диапазоне частот 0,1–2 ГГц с относительной погрешностью не более 12 %.

Моделирование одиночной микрополосковой линии в диапазоне параметров. В процессе тестирования системы TALGAT проведен квазистатический анализ конфигурации одиночной микрополосковой линии для случая без влагозащитного покрытия (рис. 3.20). В конфигурации с помощью дуг учитывается форма поперечного сечения реальной печатной трассы.

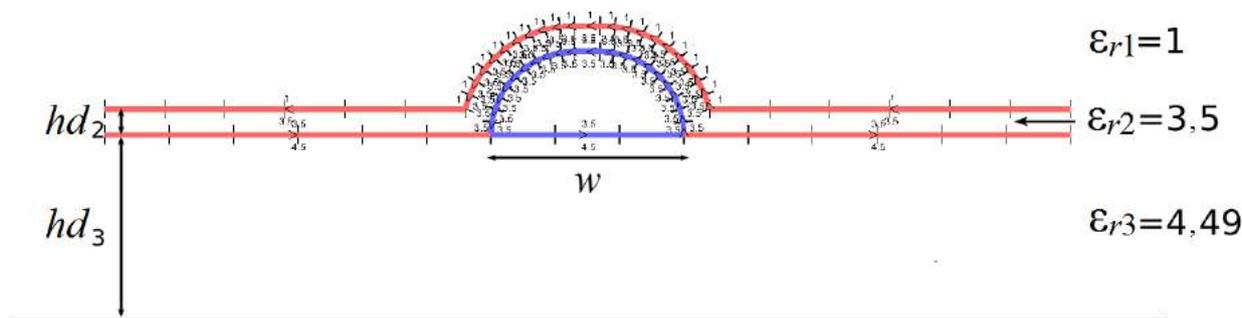


Рис. 3.20. Конфигурация одиночной микрополосковой линии

Конфигурация на рис. 3.20 включает 5 параметров: ширину проводника w , толщину паяльной маски hd_2 , толщину препрега hd_3 , диэлектрические проницаемости ϵ_{r2} и ϵ_{r3} . В качестве исходных значений использовались данные для трассы печатной платы радиотехнического блока БА КА: $w=0,75$ мм, $hd_2=0,02$ мм, $hd_3=0,1446$ мм, $\epsilon_{r2}=3,5$, $\epsilon_{r3}=4,49$.

Цель анализа микрополосковой линии – определить значения параметров, которые позволяют получить оптимальное волновое сопротивление, на примере плеча делителя мощности: $Z_{opt} = \sqrt{2} \cdot 50$ Ом. Для этого необходимо выполнить расчет волнового сопротивления Z с помощью модуля двумерного квазистатического анализа. Погрешность расчета зависит от выбранной сегментации.

Выбор длины сегмента. Реализованный в модуле квазистатического анализа MOM2D метод моментов, как и любой численный метод, имеет погрешность вычисления. Пользователь может уменьшить эту погрешность посредством выбора более частой сегментации границ входной конфигурации, задав длину сегмента с помощью команды SET_AUTO_SEGMENT_LENGTH. Однако это увеличивает общее количество сегментов и, следовательно, время вычисления Z , так как количество сегментов во входной

конфигурации определяет порядок матрицы системы линейных алгебраических уравнений, которая решается в методе моментов. Чем больше порядок матрицы, тем больше необходимо свободной памяти и процессорного времени для решения СЛАУ. Поэтому следует выбирать такой размер сегментов, который позволяет получить требуемую точность и в то же время достаточно быстро выполнить вычисления.

Допустим, что в данной задаче анализа достаточно получить погрешность вычислений менее 1 %. Примем в качестве истинного значения $Z_{ист} = 24,28$ Ом, вычисленное для исходных параметров w , hd_2 , hd_3 , ϵ_{r2} , ϵ_{r3} при очень частой сегментации 0,00625 мм (1342 сегмента). Тогда погрешности 1 % соответствует ошибка 0,2428 Ом. Затем будем увеличивать размер сегментов в 2 раза и вычислять абсолютную и относительные погрешности полученных значений Z (табл. 3.2).

Таблица 3.2

Абсолютные и относительные погрешности вычисления волнового сопротивления

segm, мм	Z , Ом	$Z - Z_{ист}$, Ом	$(Z - Z_{ист}) / Z_{ист}$, %
0,00625	24,28	0	0
0,0125	24,29	0,01	0,04
0,025	24,32	0,04	0,20
0,05	24,45	0,17	0,71
0,10	24,71	0,44	1,82

Из табл. 3.2 видно, что при сегментации 0,1 мм относительная погрешность расчета составляет 1,82 %, что превышает заданную границу 1 %. Поэтому для дальнейших вычислений целесообразно использовать сегментацию 0,05 мм.

Анализ в диапазоне параметров. Первый этап анализа микрополосковой линии состоит в определении степени влияния отдельных параметров на значение Z . Для этого будем варьировать один параметр микрополосковой линии (w , hd_2 , hd_3 , ϵ_{r2} или ϵ_{r3}) в заданном диапазоне и вычислять Z , присваивая остальным параметрам фиксированные исходные значения ($w=0,75$ мм, $hd_2=0,02$ мм, $hd_3=0,1446$ мм, $\epsilon_{r2}=3,5$, $\epsilon_{r3}=4,49$).

Так как целью варьирования параметров является получение волнового сопротивления Z , близкого к оптимальному, то введем целевую функцию

$$F(\mathbf{y}) := |Z(\mathbf{y}) - Z_{\text{opt}}|, \quad (3.2)$$

где \mathbf{y} – вектор параметров (в данном случае состоящий из одного элемента – варьируемого параметра микрополосковой линии); $Z(\mathbf{y})$ – вычисленное для \mathbf{y} волновое сопротивление; Z_{opt} – оптимальное волновое сопротивление ($Z_{\text{opt}} = \sqrt{2} \cdot 50$ Ом). Целевая функция $F(\mathbf{y})$ равна нулю, если $Z(\mathbf{y}_{\text{opt}}) = Z_{\text{opt}}$, где \mathbf{y}_{opt} – оптимальный вектор параметров.

Вычислим волновое сопротивление для параметров w (при фиксированном $hd_3=0,02$ мм) и hd_3 (при фиксированном $w=0,75$ мм), варьируя каждый из них в диапазоне 0,05; 0,06; ...; 1 мм. Затем получим значения целевой функции путем вычитания из Z оптимального значения Z_{opt} . Для сравнения выполним те же вычисления для hd_2 . Полученные графики $F(w)$ (сплошная линия), $F(hd_2)$ (линия в виде насечек) и $F(hd_3)$ (пунктирная линия) представлены на рис. 3.21.

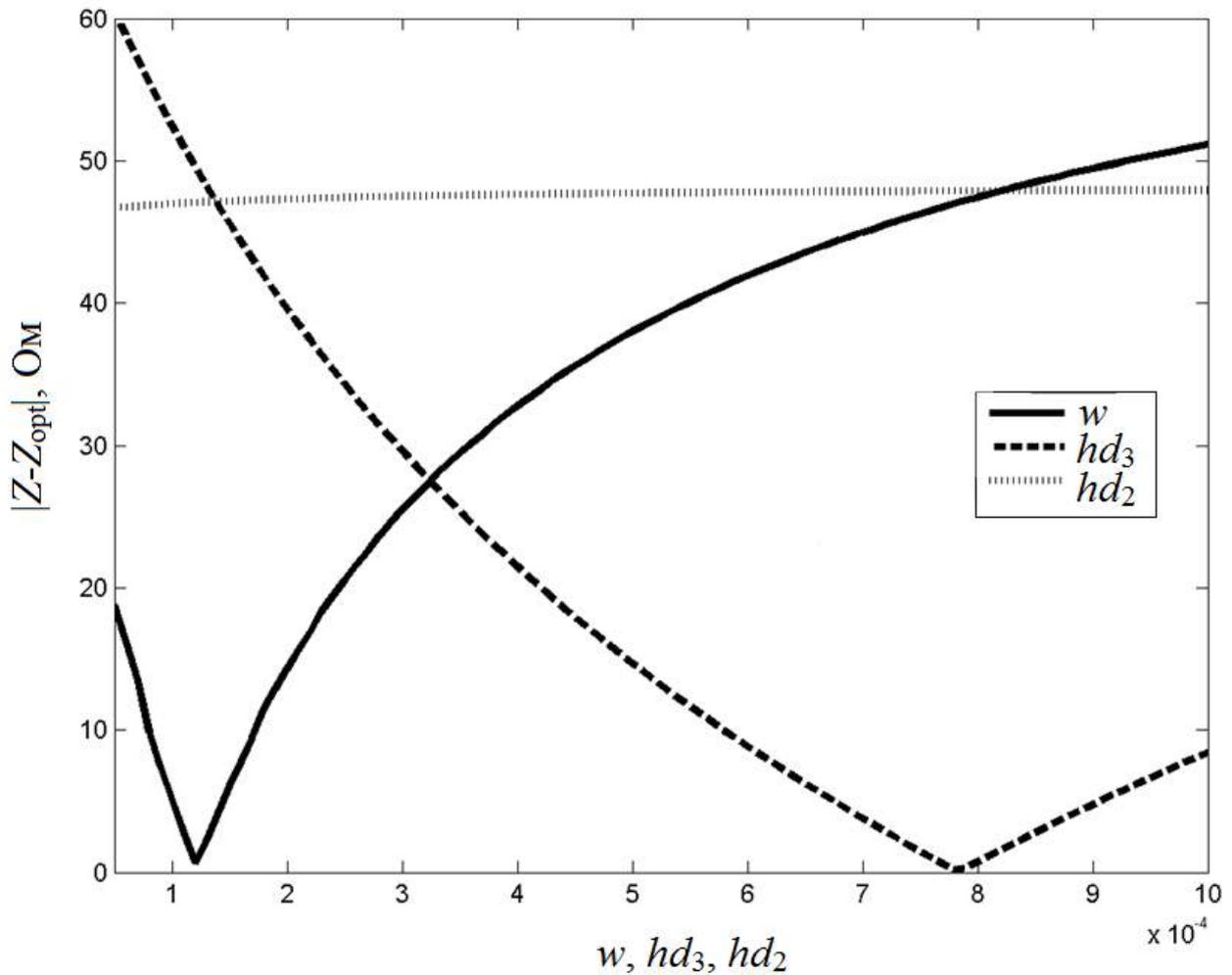


Рис. 3.21. Зависимость функции качества от параметров w , hd_2 и hd_3

Как видно из рис. 3.21, правильный подбор w или hd_3 позволяет получить оптимальное волновое сопротивление. Так, при $w=0,12$ мм (минимум на сплошной линии) значение Z будет отличаться от Z_{opt} на 0,65 Ом, а при $hd_3=0,78$ мм (минимум на пунктирной линии) – на 0,14 Ом. Однако изменение параметра hd_2 не оказывает значительного влияния на Z . Невозможно достичь Z_{opt} , выбрав какое-либо значение hd_2 из рассматриваемого на рис. 3.21 диапазона, поэтому график $F(hd_2)$ не имеет экстремальной точки в отличие от графиков $F(w)$ и $F(hd_3)$.

Аналогичная ситуация наблюдается для параметров ϵ_{r2} и ϵ_{r3} (рис. 3.22): волновое сопротивление зависит от них незначительно и не может быть приближено к Z_{opt} выбором тех или иных значений ϵ_{r2} и ϵ_{r3} при фиксированных w , hd_2 и hd_3 .

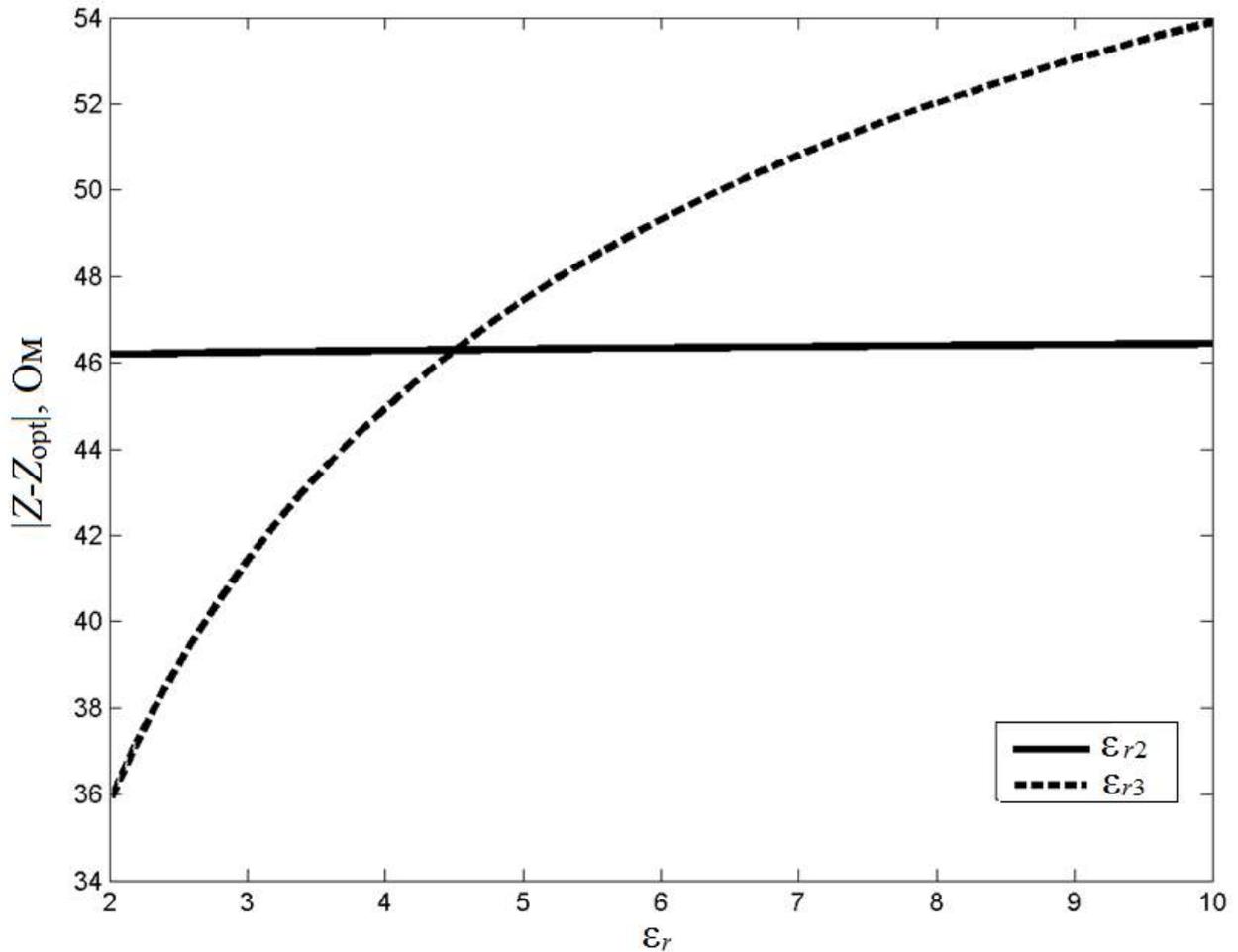
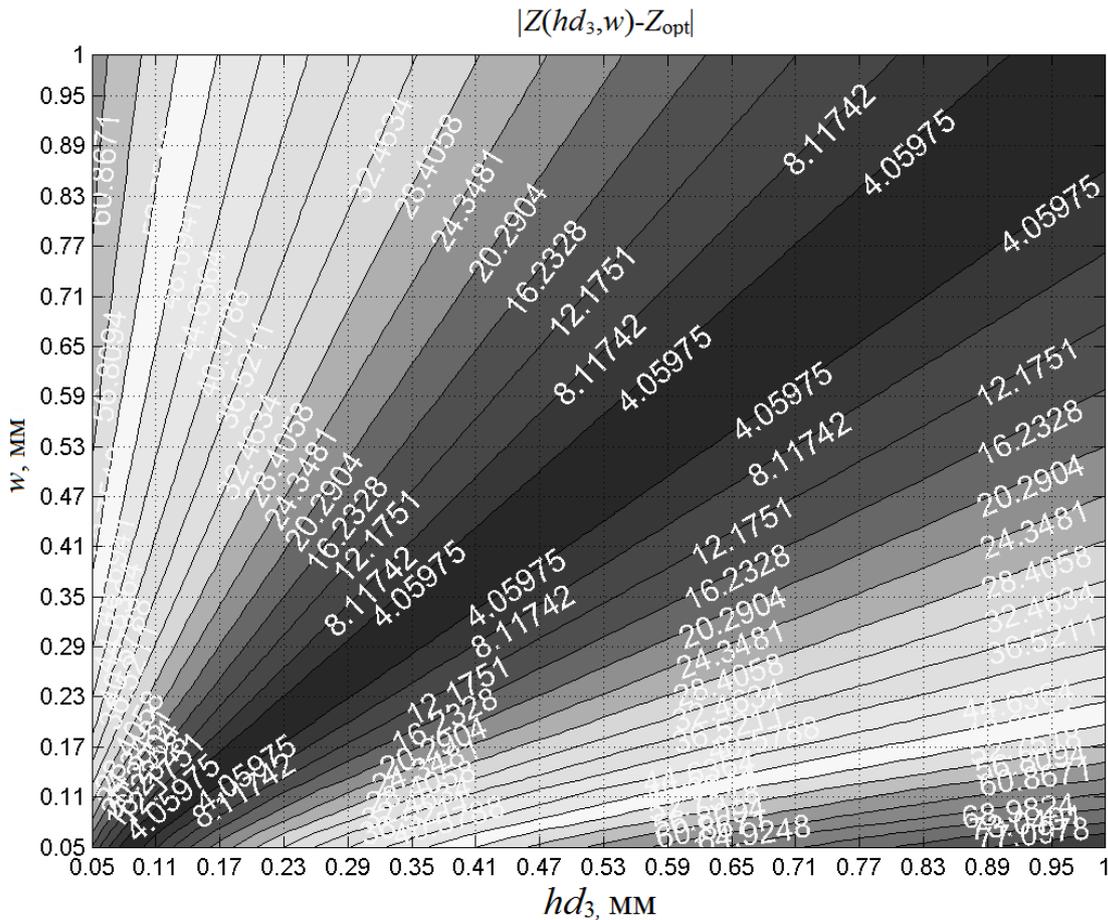


Рис. 3.22. Зависимость функции качества от параметров ϵ_{r2} и ϵ_{r3}

Таким образом, значение волнового сопротивления и, следовательно, целевой функции определяется в первую очередь значениями параметров w и hd_3 , то есть шириной проводника и толщиной препрега.

Зависимость волнового сопротивления от w и hd_3 . Рассмотрим более подробно зависимость волнового сопротивления от параметров w и hd_3 путем одновременного их варьирования в диапазоне 0,05; 0,06; ...; 1 мм. Полученные значения целевой функции $F(w, hd_3)$ представлены в виде контурного графика на рис. 3.23, где изолинии проведены по точкам, в которых значения $F(w, hd_3)$ равны написанному на изолинии числу. Например, две проходящие вблизи диагонали изолинии имеют надпись 4,05975. Это означает, что для всех точек, которые лежат между указанными изолиниями, значение целевой функции $F(w, hd_3)$ не превышает 4,05975 Ом, то есть волновое сопротивление отличается от Z_{opt} не более чем на 4 Ом.



Из рис. 3.23 видно, что целевая функция $F(w, hd_3)$ имеет вблизи диагонали между изолиниями 4,05975 область минимальных значений, которая расширяется при росте параметров w и hd_3 . Данный факт говорит о том, что при увеличении линейных размеров одиночной микрополосковой линии чувствительность волнового сопротивления к изменениям параметров w и hd_3 снижается. Следовательно, необходимо выбирать большие значения w и hd_3 , так как при этом снижаются требования к точности изготовления микрополосковой линии.

По контурному графику на рис. 3.23 можно определить оптимальные сочетания параметров w и hd_3 . Зададим, например, ширину проводника $w=0,53$ мм и найдем такое значение толщины препрега hd_3 , чтобы волновое сопротивление отличалось от Z_{opt} не более чем на 4 Ом. Для этого найдем на оси ординат значение 0,53, проведем от него параллельную оси абсцисс линию (в данном случае эта линия уже проведена на рис. 3.23 как линия сетки) и найдем точки, где построенная линия пересекает изолинии со значениями $F(w, hd_3)$, близкими к 4 Ом. На рис. 3.23 такими изолиниями являются изолинии 4,05975, построенная линия пересекает их в точках (0,5; 0,53) и (0,63; 0,53). Следовательно, для получения волнового сопротивления Z , которое отличается от Z_{opt} не более чем на 4 Ом, нужно при $w=0,53$ мм выбрать толщину препрега hd_3 в диапазоне от 0,5 до 0,63 мм. При этом наиболее близкое к Z_{opt} волновое сопротивление можно получить, если выбрать hd_3 из середины диапазона.

Зависимость волнового сопротивления от w , ϵ_{r3} и hd_3 , ϵ_{r3} . Кроме целевой функции $F(w, hd_3)$, интерес представляет также зависимость Z от параметров w , ϵ_{r3} и hd_3 , ϵ_{r3} .

Для построения контурных графиков значение ϵ_{r3} варьировалось в диапазоне 1; 1,1; ...; 10. Контурный график для целевой функции $F(w, \epsilon_{r3})$ показан на рис. 3.24.

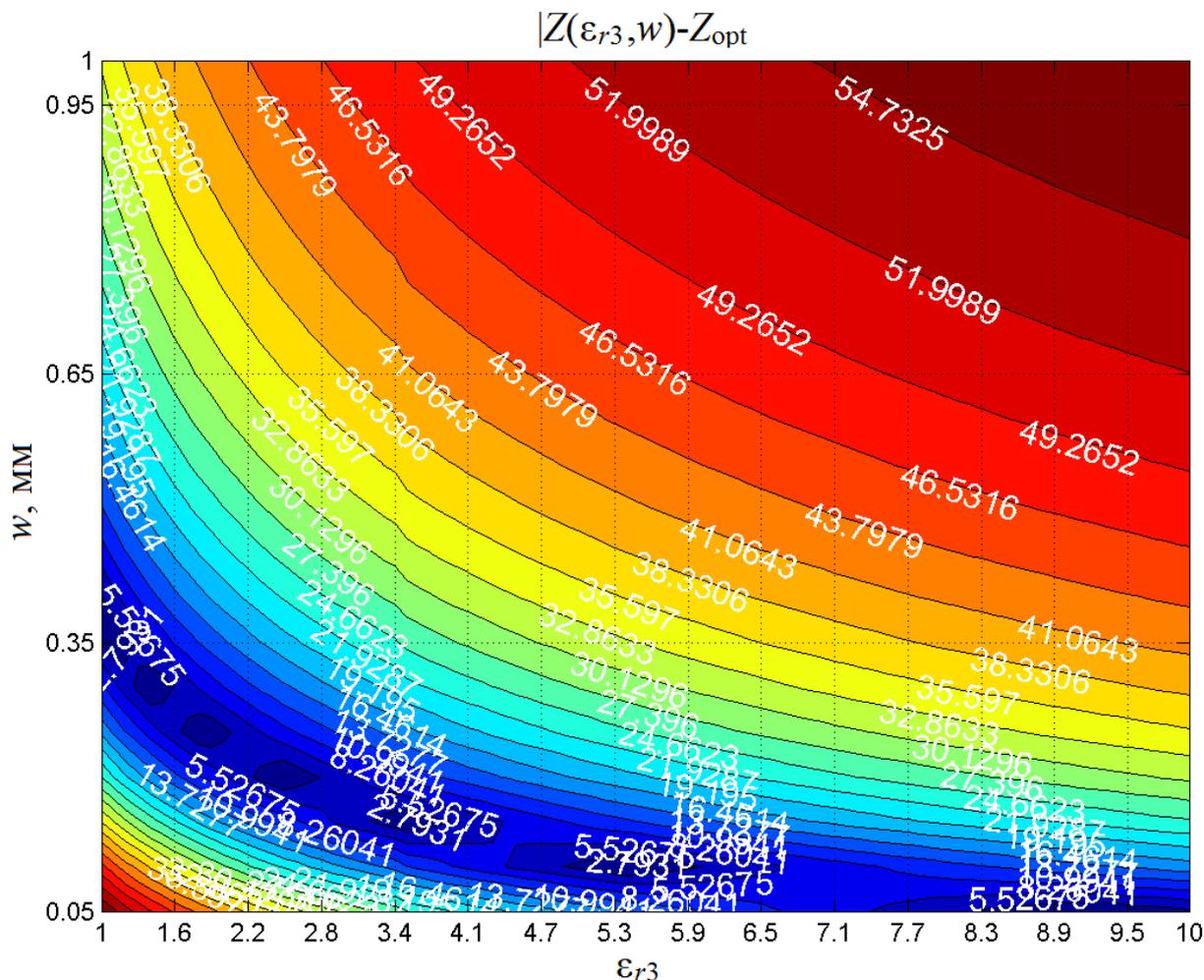


Рис. 3.24. Зависимость целевой функции от параметров w и ϵ_{r3}

Изолинии на рис. 3.24 образуют контурный график, который отличается от графика на рис. 3.23. Область минимальных значений волнового сопротивления здесь лежит не на диагонали, а в нижней половине рисунка. Это говорит о том, что если ширина проводника больше 0,5 мм, то изменением параметра ϵ_{r3} (подбором материала диэлектрика) невозможно получить оптимальное волновое сопротив-

ление Z_{opt} . Однако при $w < 0,5$ мм имеется возможность получить волновое сопротивление, которое отличается от Z_{opt} менее чем на 2,8 Ом.

Контурный график для целевой функции $F(hd_3, \epsilon_{r3})$ (рис. 3.25) подобен контурному графику $F(w, hd_3)$ с тем отличием, что область малых значений целевой функции находится дальше от диагонали рисунка. По данному графику можно определить оптимальные сочетания параметров hd_3 и ϵ_{r3} , то есть по заданной толщине препрега выбрать материал диэлектрика и наоборот.

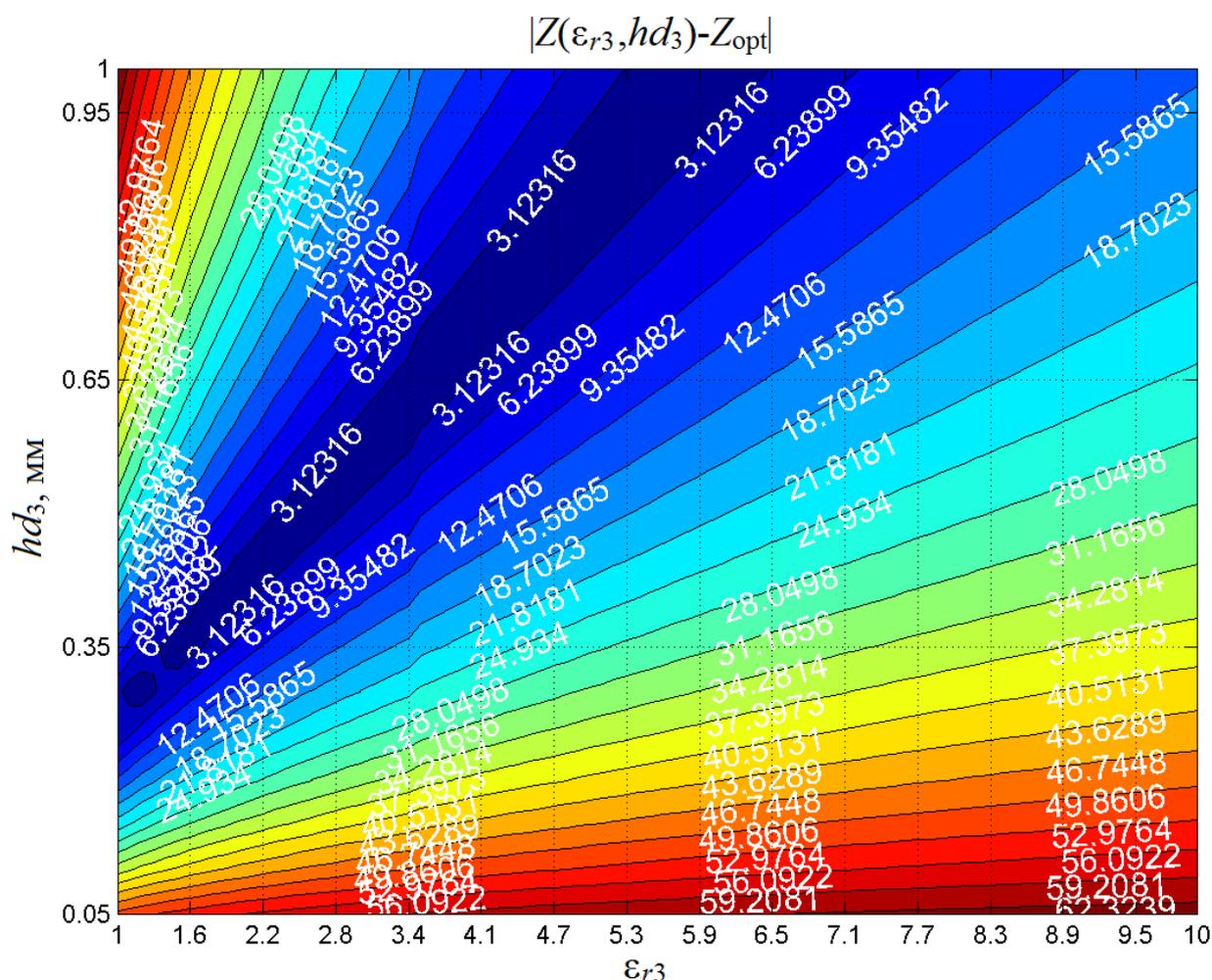


Рис. 3.25. Зависимость целевой функции от параметров hd_3 и ϵ_{r3}

Контурные графики для пар параметров, которые включают ε_{r2} , получены, но здесь не приведены, так как не содержат полезной информации: либо минимальное достигаемое значение Z отличается от Z_{opt} больше чем на 40 Ом (целевая функция $F(hd_2, \varepsilon_{r2})$), либо контурные графики отражают зависимость только от геометрического параметра (целевые функции $F(hd_3, \varepsilon_{r2})$ и $F(w, \varepsilon_{r2})$).

Статистический анализ по методу Монте-Карло: расчет наихудшего случая. Представленные выше графики позволяют оценить чувствительность волнового сопротивления к изменениям одного или двух параметров одиночной микрополосковой линии. Однако в общем случае необходимо принимать во внимание возможность отклонения реальных значений от расчетных сразу для нескольких параметров одиночной линии, например в результате неточности изготовления или воздействия внешней среды. Статистический анализ по методу Монте-Карло позволяет учесть возможность таких отклонений.

Метод Монте-Карло состоит в следующем: для заданного диапазона изменения параметров генерируются случайным образом наборы параметров, для которых вычисляются значения волнового сопротивления Z . После этого проводится обработка полученных значений Z . Выполним расчет наихудшего случая, то есть выберем такой случайно сгенерированный набор параметров, при котором Z максимально отличается от оптимального волнового сопротивления.

В качестве середины диапазона изменения параметров будем использовать указанные ранее исходные значения для микрополосковой линии: $w=0,75$ мм, $hd_2=0,02$ мм, $hd_3=0,1446$ мм, $\varepsilon_{r2}=3,5$, $\varepsilon_{r3}=4,49$. В соответствии с опублико-

ванными данными [97] разброс параметров печатных плат на этапе производства составляет в среднем 2,3 % и более. Для учета такого разброса зададим нижние и верхние границы диапазона изменения параметров с запасом: $\pm 10\%$ относительно середины диапазона. Таким образом, для параметра w получаем диапазон 0,675; ... ; 0,825 мм, для параметра hd_2 – 0,018; ... ; 0,022 мм, и так далее. Для генерации случайных чисел будем использовать равномерное распределение. Результаты экспериментов приведены в табл. 3.3 где в первом столбце указано количество случайно сгенерированных наборов параметров, во втором – среднее значение волнового сопротивления Z_{avg} для всех вычисленных Z , в третьем – среднеквадратическое отклонение σZ , в четвертом и пятом столбцах – минимальное и максимальное Z из всех вычисленных, а в шестом – относительная ошибка.

Таблица 3.3

Статистический анализ волнового сопротивления по методу Монте-Карло

Кол-во	Z_{avg} , Ом	σZ , Ом	Z_{min} , Ом	Z_{max} , Ом	$\sigma Z / Z_{avg}$, %
100	24,56	1,61	21,06	28,06	6,56
1000	24,58	1,57	20,39	29,11	6,39
10000	24,53	1,66	20,13	29,49	6,77

На основании данных табл. 3.3 можно сделать вывод, что при изменении значений параметров одиночной микрополосковой линии в пределах $\pm 10\%$ от исходных среднеквадратическое отклонение волнового сопротивления от значения Z , вычисленного для исходных значений параметров, не превышает 7 %. Кроме того, при изменении параметров микрополосковой линии в заданном диапазоне волновое сопротивление в наихудшем случае может отклоняться.

Результаты моделирования. Полученные двумерные графики и контурные графики позволяют определить оптимальные значения параметров. Следующим шагом является автоматизация процедуры подбора оптимальных параметров с применением методов оптимизации. Учитывая небольшую размерность задачи оптимизации одиночной микрополосковой линии, целесообразно использовать алгоритм неявного фильтрования, который по сравнению с эволюционными стратегиями требует меньшего количества вычислений целевой функции для задач оптимизации с малой размерностью. Однако для многопроводных линий количество переменных растет и эволюционные стратегии могут показать для таких целевых функций гораздо большую производительность.

Моделирование дифференциальной пары в диапазоне параметров. Рассмотрим моделирование дифференциальной пары на печатной плате БА КА с целью выбора из заданного диапазона параметров такой их комбинации, при которой отклонение искомой величины (волнового сопротивления) от требуемого значения ($Z_{opt}=50 \text{ Ом}$) минимально. С помощью команд системы TALGAT формируется конфигурация дифференциальной пары и вычисляется волновое сопротивление. Затем производится построение контурного графика зависимости волнового сопротивления от геометрических и физических параметров пары. На рис. 3.26 приведено поперечное сечение рассматриваемой дифференциальной пары. Конфигурация на рис. 3.26 имеет 8 параметров, это ширина проводника w , высота проводника t , расстояние между проводниками s , расстояние от края до проводника d и т.д.

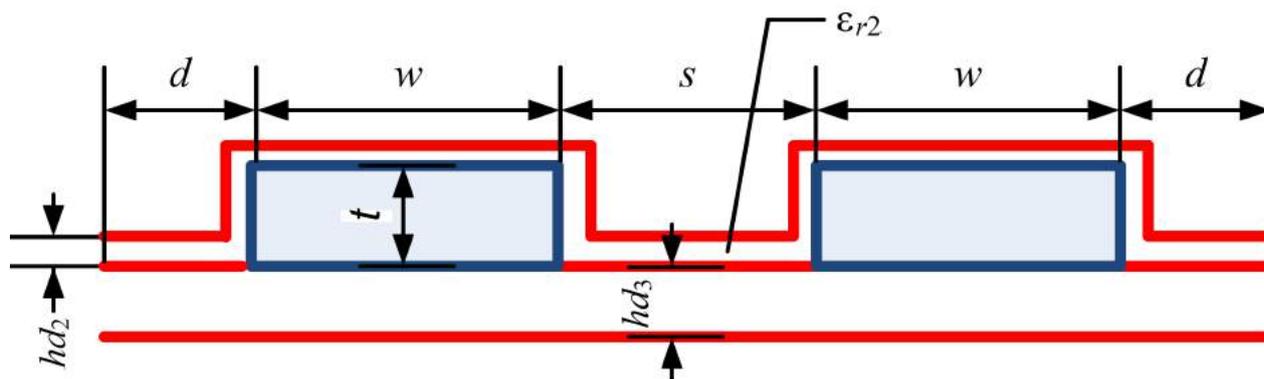
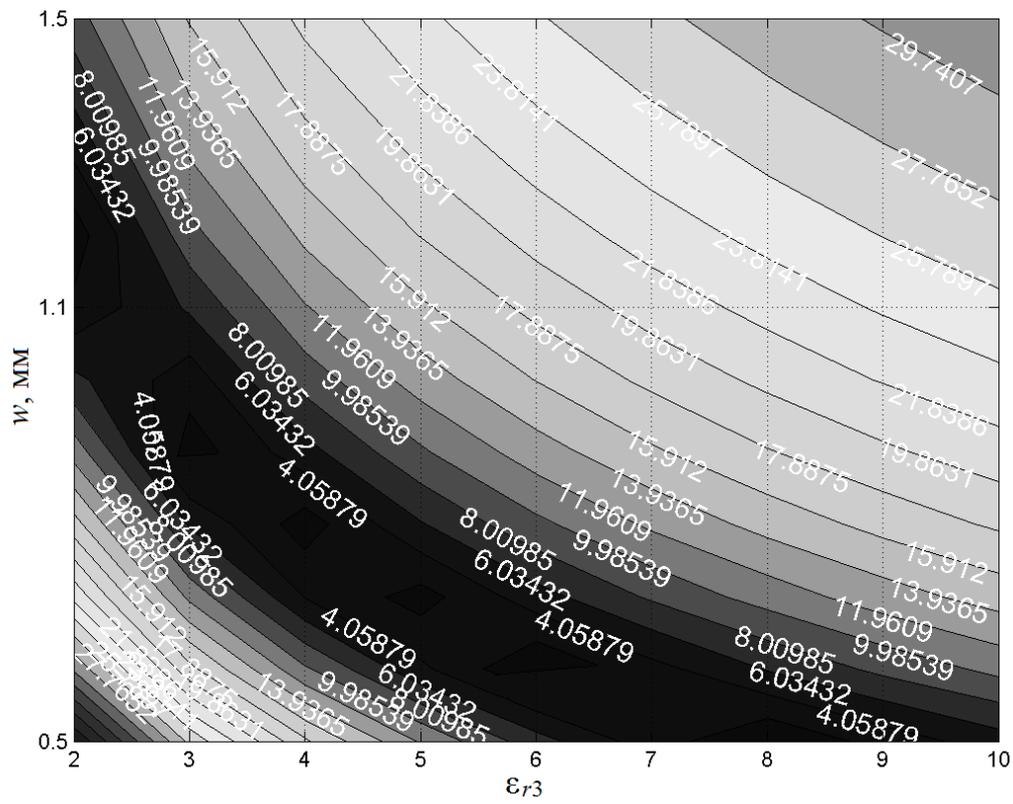


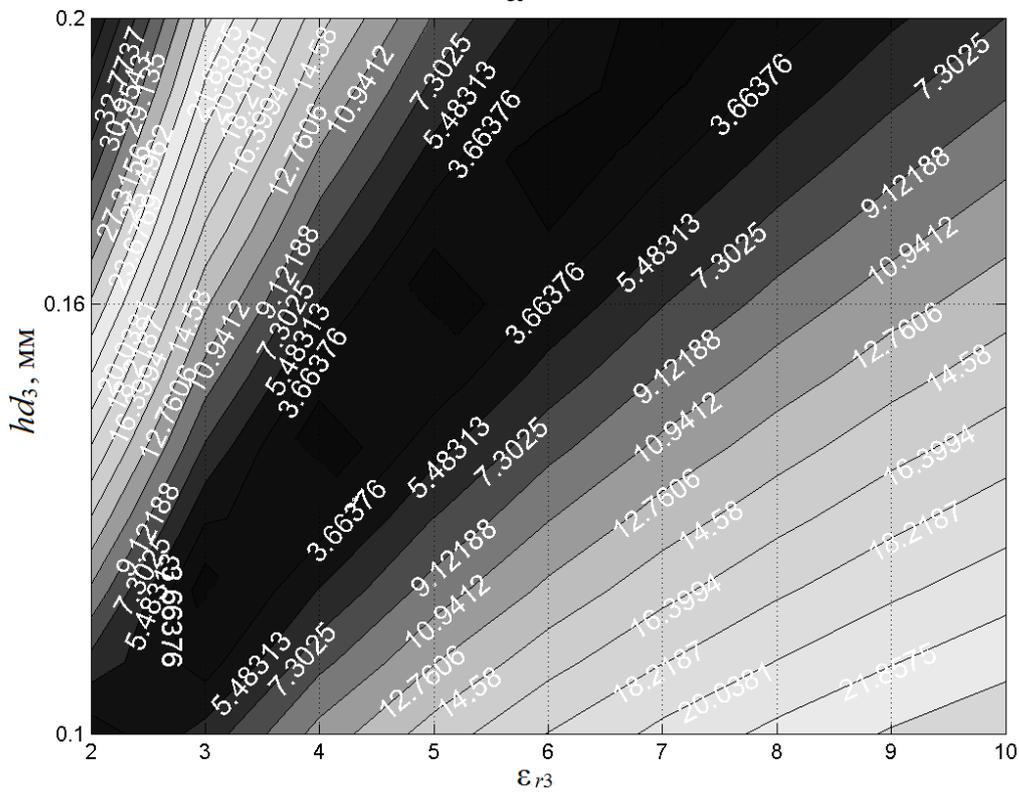
Рис. 3.26. Конфигурация дифференциальной пары

После проведения вычислительных экспериментов для исследования выбрали 3 параметра, которые оказывают наибольшее влияние на волновое сопротивление: ширину проводника w , толщину препрега hd_3 , диэлектрическую проницаемость ϵ_{r3} . Значения параметров, которые не варьировались, — $\epsilon_{r2}=3,5$, $hd_2=20$ мкм, $t=0,065$ мм, $s=1,5$ мм, $d=1$ мм. Было выбрано разбиение границ исследуемой структуры на 64 сегмента, так как при этом отклонение значения Z от $Z_{\text{ист}}$ ($Z_{\text{ист}}=50,1585$ Ом — значение, полученное при числе сегментов 128) составляет 0,14 %.

Введена целевая функция (3.2). Параметры дифференциальной пары варьировались в следующем диапазоне: $w=0,1; 0,11; \dots; 1,5$ мм, $hd_3=0,01; 0,011; \dots; 0,2$ мм, $\epsilon_{r3}=2; 3; \dots; 10$. Полученные контурные графики зависимости целевой функции $F(y)$ от параметров w и ϵ_{r3} , hd_3 и ϵ_{r3} показаны на рис. 3.27. На рис. 3.27, а видно, что целевая функция $F(w, \epsilon_{r3})$ имеет в нижней части между изолиниями 4,0587 область минимальных значений. Это говорит о том, что при $w < 1,2$ мм можно получить волновое сопротивление, которое отличается от Z_{opt} менее чем на 4 Ом. Если ширина проводника больше 1,2 мм, то изменением параметра ϵ_{r3} невозможно добиться Z_{opt} .



a



б

Рис. 3.27. Зависимость целевой функции от параметров w и ϵ_{r3} (*a*), hd_3 и ϵ_{r3} (*б*)

Оптимальную комбинацию параметров для рис. 3.27, а можно получить следующим образом: зададим ширину проводника 0,75 мм и найдем такое значение диэлектрической проницаемости ϵ_{r3} , чтобы волновое сопротивление отличалось от Z_{opt} не более чем на 4 Ом. Для этого на оси ординат отметим значение 0,75 мм, проведем от него параллельную оси абсцисс линию и найдем точки, где построенная линия пересекает изолинии со значениями $F(w, \epsilon_{r3})$, близкими к 4 Ом. Такими изолиниями являются изолинии 4,05879. Построенная линия пересекает их в точках (3; 0,75) и (4,5; 0,75), следовательно, величину диэлектрической проницаемости препрега нужно выбирать в диапазоне от 3 до 4,5, при этом наиболее близкое к Z_{opt} значение Z можно получить, выбрав ϵ_{r3} из середины диапазона.

Проанализировав аналогично второй график и полученные значения параметров hd_3, ϵ_{r3} , можно подобрать их комбинацию, при которой отклонение волнового сопротивления от Z_{opt} будет минимальным.

Таким образом, на основе вычисленных данных подготовлены контурные графики зависимости волнового сопротивления Z от геометрических и физических параметров дифференциальной пары. С помощью данных графиков можно выбрать параметры дифференциальной пары, которые соответствуют минимальному отклонению Z от требуемого значения.

3.7. Программная реализация модуля эволюционных стратегий

Рассмотрим реализацию ЭС [75, 103, 104] в виде динамически загружаемого модуля, который пользователь может подключить командой «INCLUDE ES». Особенность системы TALGAT состоит в открытости для включения новых моделей анализа разнотипных структур, что не позволяет «жестко» закодировать определенный набор целевых функций. Однако наличие встроенного в ядро системы интерпретатора команд позволяет решить эту проблему.

Особым элементом реализации модуля ЭС в системе является целевая функция – созданная пользователем команда на языке скриптов системы. Целевая функция хранится интерпретатором и при необходимости может быть вычислена с помощью алгоритмов анализа. При этом значения параметров целевой функции внутри пользовательской команды являются переменными, которые пользователь может использовать по своему усмотрению (например, в качестве одного из геометрических параметров структуры для его оптимизации).

Работа модуля ЭС состоит из следующих этапов.

1. Настройка и запуск ЭС специальными командами (ES_MIN, ES_MAX, LOPT_ES_MIN, LOPT_ES_MAX).
2. Запрос значения целевой функции для каждой новой точки поиска.
3. Интерпретация целевой функции и вызов соответствующих алгоритмов анализа.
4. Возврат значения целевой функции.

5. Если условие останова ЭС (например, заданное количество итераций) не достигнуто – переход на этап 2.

Описываемая в данной работе реализация имеет ряд достоинств.

1. Пользователю предоставлен универсальный метод оптимизации, совместимый со всеми моделями анализа, которые имеются в системе на данный момент или появятся в будущем.
2. Возможна оптимизация любых параметров структур [105].
3. Для формирования целевой функции могут использоваться любые вычисляемые системой характеристики структур.
4. Для работы с ЭС не требуется каких-либо модификаций вычислительных модулей системы.

К недостаткам модуля ЭС можно отнести затраты на интерпретацию целевой функции для каждой особи во время работы ЭС, однако ими можно пренебречь, учитывая простоту интерпретатора и значительные вычислительные затраты самого анализа структур.

Скриптовые команды. После загрузки в систему TALGAT модуля ЭС пользователю становится доступна команда ES_MIN, при вызове которой выполняется алгоритм $(\mu/\mu_I, \lambda)$ - σ -самоадаптационной ЭС. Команда ES_MIN имеет следующие параметры.

1. Размер популяции родителей μ (рекомендуется $\mu = \lambda/3$).

2. Размер популяции потомков λ .
3. Количество поколений $nGens$.
4. Размерность пространства поиска N (количество оптимизируемых параметров целевой функции).
5. Имя динамической команды, которая реализует целевую функцию, принимает N параметров и возвращает вещественное число.

Для задания дополнительных настроек оптимизации предназначена команда `ES_OPTS`, которая принимает следующие параметры.

1. Параметр самообучения τ (по умолчанию $\tau = 1/\sqrt{N}$, если ввести $\tau = 0$, то команда `ES_OPTS` автоматически вычислит τ по этой формуле).
2. Начальное значение величины мутации σ (рекомендуется выбирать равным $1/3$ диапазона допустимых значений параметров целевой функции).
3. Начальная точка поиска (зависит от целевой функции).

Для получения результатов оптимизации используются команды `GET_BEST_ES_RESULT` и `GET_BEST_ES_PARAMETER`.

Пример оптимизации ЭС целевой функции приведен в листинге 3.4.

Алгоритм выполнения скриптовых команд. Схема алгоритма выполнения скриптовых команд модуля ЭС представлена на рис. 3.28. Пользователь запускает оптимизацию путем ввода команды `ES_MIN`, после чего интерпретатор находит обработчик команды `ES_MIN` и вызывает его.

Листинг 3.4 Пример оптимизации ЭС целевой функции

```
INCLUDE "ES"
INCLUDE "UTIL"
INCLUDE "MATRIX"
INCLUDE "INFIX"

SET "nGens" 10
SET "aN" 10

SET "y" CREATE_REAL_MATRIX 1 aN // simple sphere
fitness function
CREATE_KEYWORD "Sphere"
SET "y" SET_MATRIX_ROW y 0
GA_PARAM_1 GA_PARAM_
GA_PARAM_3 GA_PARAM_4
GA_PARAM_5 GA_PARAM_6
GA_PARAM_7 GA_PARAM_8
GA_PARAM_9 GA_PARAM_10
SET "sum" 0.
FOR "i" 0 MINUS aN 1 1 SET "sum" PLUS sum MUL
GET_MATRIX_VALUE y 0 i GET_MATRIX_VALUE y 0 i
END_CREATE_KEYWORD SQRT sum

ES_OPTS 0 1 10.
ECHO TIMER ES_MIN 3 10 nGens aN "Sphere"
ECHO GET_BEST_ES_RESULT
ECHO GET_BEST_ES_PARAMETER 0
ECHO GET_BEST_ES_PARAMETER MINUS aN 1
```

Обработчик считывает заданные пользователем параметры и использует их для инициализации параметров ЭС.

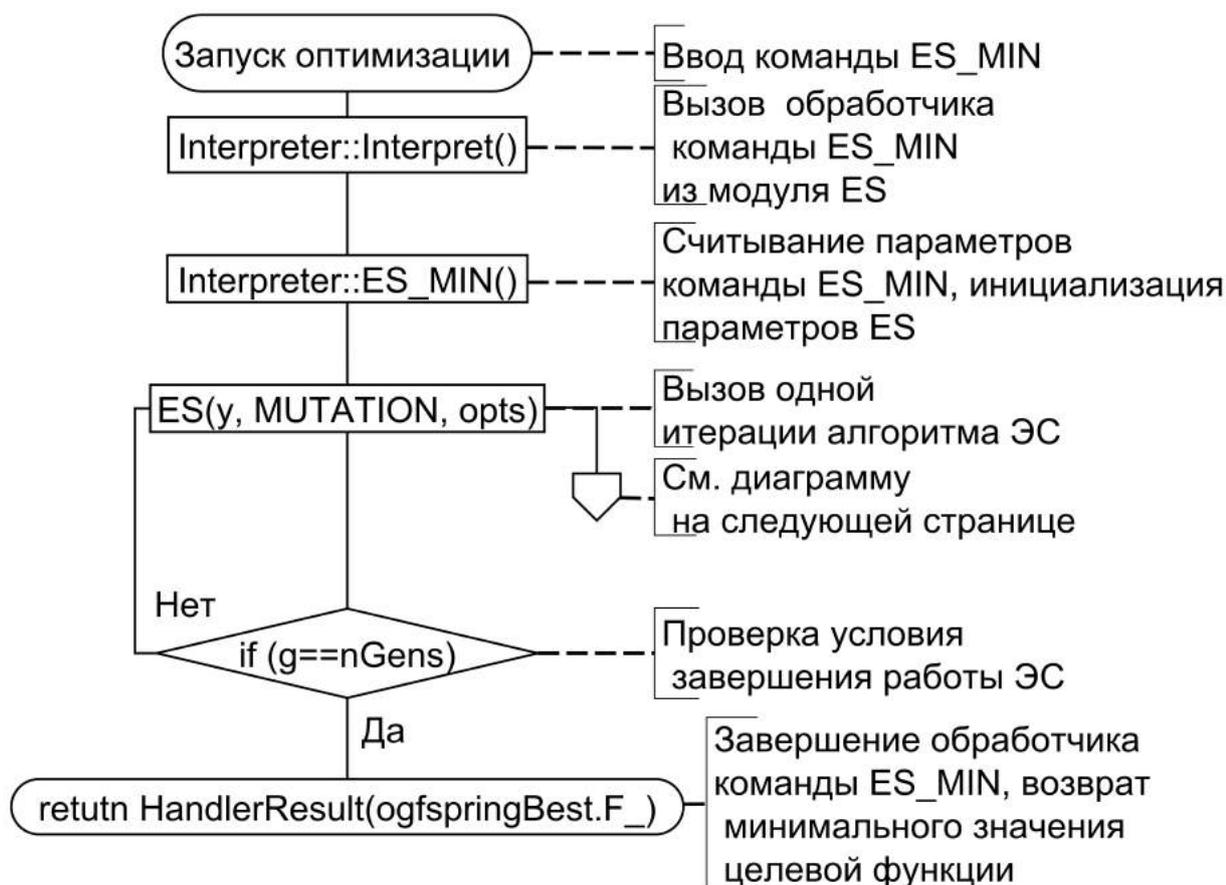


Рис. 3.28. Схема алгоритма выполнения скриптовых команд модуля ЭС

Затем обработчик вызывает функцию $ES()$, которая реализует одну итерацию (одно поколение) алгоритма ЭС. После этого выполняется проверка условий завершения работы ЭС (на рис. 3.28 в качестве примера приведена проверка на выполнение заданного количества поколений $nGens$). Если ни одно из условий не выполняется, алгоритм снова вызывает функцию $ES()$. Иначе обработчик завершает работу и возвращает минимальное значение целевой функции. При этом оптимальное значение параметров целевой функции сохраняется и может быть затем прочитано с помощью команды `GET_BEST_ES_PARAMETER`.

Алгоритм вычисления целевой функции. Рассмотрим реализацию функции $ES()$, схема алгоритма которой приведена на рис. 3.29.



Рис. 3.29. Схема алгоритма функции $ES()$

Функция $ES()$ вызывается для каждой итерации алгоритма ЭС. В соответствии с алгоритмом ЭС (см. алгоритм 1.1) функция $ES()$ создает λ потомков путем прибавления к текущей точке поиска умноженного на интенсивность мутации нормально распределенного случайного

числа, которое генерируется функцией `randn()`. После этого выполняется проверка переменной `opts.iParallelThreads`, значение которой задается пользователем с помощью команды `SET_ES_THREADS`. Если значение `opts.iParallelThreads` больше единицы, то вызывается параллельная версия целевой функции `ParallelF()`, которая рассмотрена ниже. Иначе вычисление значений целевой функции выполняется последовательно в одном потоке следующим образом: для каждого потомка вызывается функция `opts.F()`, которая в качестве параметра принимает вектор параметров l -го потомка `offspring[l].y_`. Функция `opts.F()` вызывает функцию `RefreshVariables()`, которая с помощью метода `SetVariable()` класса `Interpreter` создает N переменных с именами `GA_PARAM_n`, $n = 1, \dots, N$, и присваивает им значения из вектора $y_$. Затем функция `opts.F()` запускает интерпретацию целевой функции и возвращает ее значение. Данная процедура повторяется для каждого потомка, которые затем сортируются по возрастанию значений целевой функции. В конце функции `ES()` вычисляются значения новой текущей точки поиска и новой интенсивности мутации.

Алгоритм параллельного вычисления целевой функции. Рассмотрим схему алгоритма функции `ParallelF()` (рис. 3.30), которая позволяет вычислять значения целевой функции для каждого потомка в отдельном потоке. В начале функции `ParallelF()` происходит считывание целевой функции в каждый из интерпретаторов (количество интерпретаторов равно значению переменной `opts.iParallelThreads`). После этого алгоритм выполняет проверку, для всех ли потомков была вычислена функция.

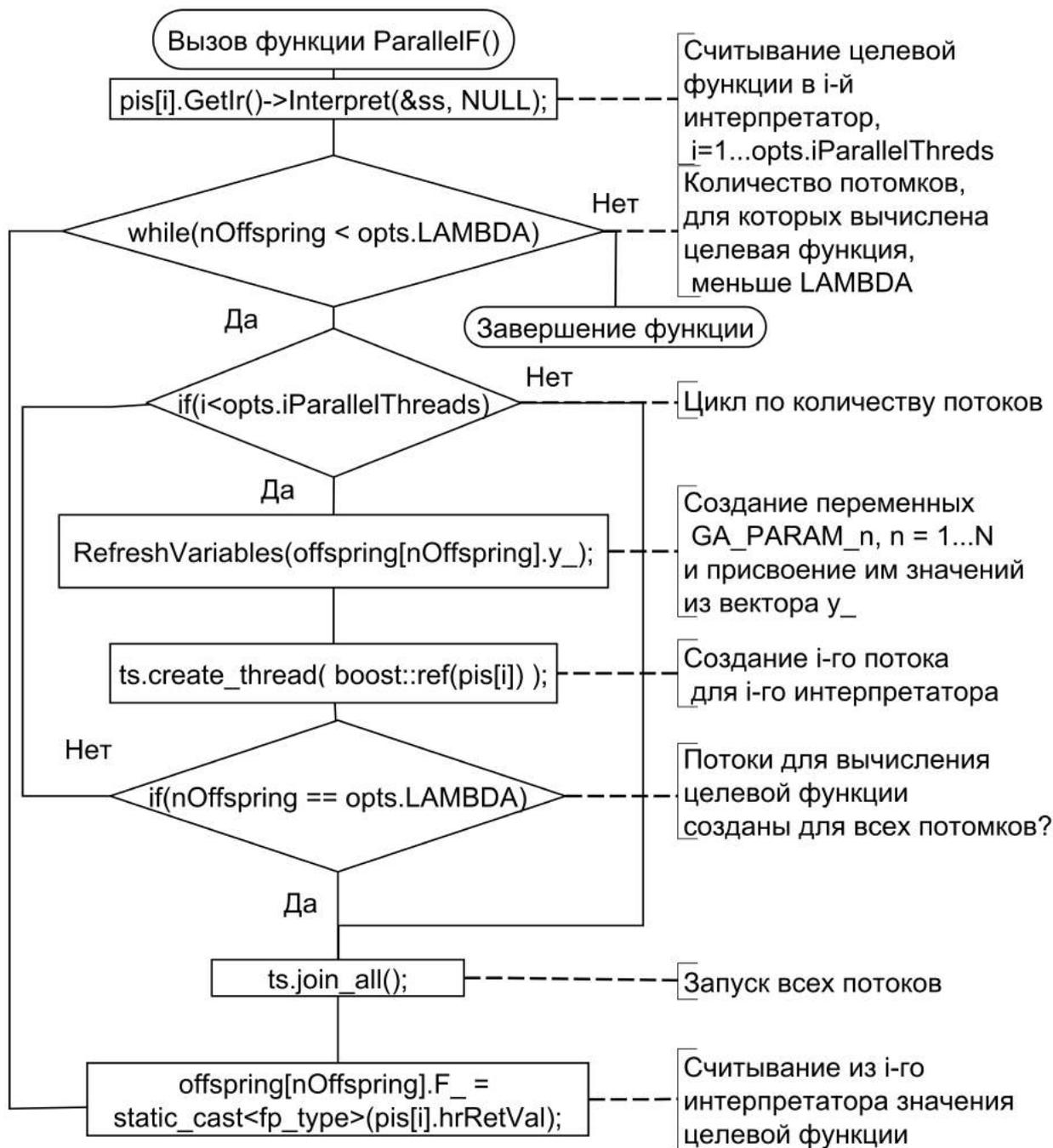


Рис. 3.30. Схема алгоритма функции ParallelF()

Если это не так, то запускается цикл по количеству потоков $\text{opts.iParallelThreads}$. Первая операция в цикле создаст в текущем i -м интерпретаторе N переменных с именами GA_PARAM_n (аналогично случаю последовательного вычисления целевой функции в одном потоке), вторая

операция с помощью метода `thread_group::create_thread()` создает объект-поток `thread` и передает ему указатель на i -й интерпретатор. После этого алгоритм проверяет, для скольких потомков были созданы потоки. Если для всех потомков потоки для вычисления целевой функции созданы, выполняется переход к операции запуска всех потоков `thread_group::join_all()`. Если потоки созданы не для всех потомков, алгоритм возвращается к началу цикла. При этом если количество созданных потоков равно значению `opts.iParallelThreads`, то алгоритм запускает имеющиеся потоки посредством операции `thread_group::join_all()`, ожидает завершения исполнения всех потоков (блокирующий вызов) и затем считывает из интерпретаторов вычисленные значения целевой функции. После этого происходит возврат к проверке количества потомков, для которых уже вычислена целевая функция. Если целевая функция вычислена не для всех потомков, повторно запускается цикл по количеству потоков `opts.iParallelThreads`. Иначе алгоритм завершается.

Алгоритм функции `ParallelF()` на рис. 3.30 обрабатывает три частных случая параллельного вычисления целевой функции.

1. Количество потомков λ равно количеству потоков `opts.iParallelThreads` (цикл по количеству потоков запускается один раз).
2. Количество потомков λ больше количества потоков `opts.iParallelThreads` (цикл по количеству потоков запускается $\lambda/\text{opts.iParallelThreads}$ раз).

3. Количество потомков λ меньше количества потоков `opts.iParallelThreads` (цикл по количеству потоков запускается один раз и завершается после λ итераций).

При этом реализация алгоритма функции `ParallelF()` в системе компьютерного моделирования ЭМС за счет повторного использования кода ядра и стандартных библиотек занимает менее 120 строк на языке программирования C++. Возможность работы кода для параллельного вычисления целевой функции обеспечивается реентерабельностью (поддержкой одновременного вызова одних и тех же функций из нескольких потоков, когда функция может быть вызвана потоком до того, как с ней завершит работу другой поток) ядра, вычислительных модулей и модулей утилит, которые безопасны при использовании в многопоточной среде.

Для проверки правильности и эффективности реализации функции `ParallelF()` выполнено ее тестирование на практической задаче оптимизации.

Тестирование алгоритма параллельного вычисления целевой функции. Для тестирования алгоритма параллельного вычисления целевой функции (см. рис. 3.30) проведен следующий вычислительный эксперимент: на компьютере с процессором Intel Core i7 970 (количество ядер – 6, частота – 3,2 ГГц, объем оперативной памяти – 24 Гбайт) запускалась оптимизация сосредоточенных нагрузок проводной антенны с помощью $(3/3_I, 10)$ - σ ЭС, в каждом запуске измерялось время, необходимое для $FEs = 100$ и 1000 . В табл. 3.4 для 1, 2 и 4 потоков приведены усредненные по 10 запускам результаты измерений времени оптимизации (в секундах, после знака \pm приведено стандартное отклонение).

Таблица 3.4

Ускорение времени оптимизации сосредоточенных нагрузок

Число потоков	$F E_s = 100$		$F E_s = 1000$	
	Время, с	Ускорение	Время, с	Ускорение
1	1291,2±0,91	1	12791±113	1
2	748,0±2,15	1,72	7290±50	1,75

Из табл. 3.4 следует, что для 100 и 1000 1,7 раза при использовании 2 потоков.

С целью сравнения аналогичный вычислительный эксперимент проведен для задачи оптимизации параметров полосковой линии для 1000 вычислений целевой функции и мелкой сегментации границ проводник-диэлектрик и диэлектрик-диэлектрик (порядок матрицы СЛАУ – 2600). Усредненные по 10 запускам результаты приведены в табл. 3.5.

Таблица 3.5

Ускорение времени оптимизации параметров полосковой линии

Число потоков	Время, с	Ускорение
1	5553,7±0,61	1
2	3004,7±0,50	1,85

Из табл. 3.4 следует, что для задачи оптимизации параметров полосковых линий применение параллельного вычисления целевой функции позволило получить выигрыш по времени в 1,85 раза по сравнению с оптимизацией в один поток.

Таким образом, при оптимизации на многоядерных процессорах целесообразно использовать параллельное вычисление целевой функции.

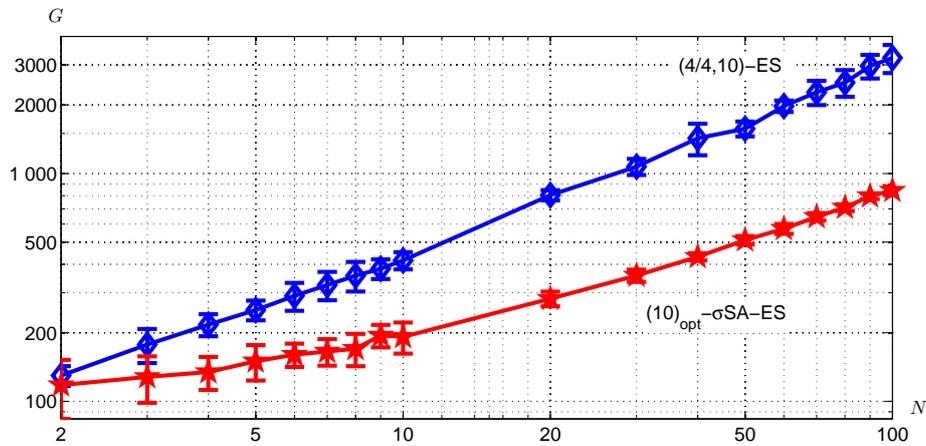
Тестирование модуля эволюционных стратегий. Для проверки правильности реализации модуля ЭС выполнено экспериментальное сравнение результатов оптимизации $(\mu/\mu_I, \lambda)$ - σ -самоадаптационной и $(\lambda)_{\text{opt}}$ - σ -самоадаптационной ЭС для тестовой целевой функции – сферической модели (1.3).

В экспериментальном сравнении (рис. 3.31) подсчитывалось количество поколений, необходимое для достижения значения целевой функции $F(y) < 10^{-10}$. Оптимизация запускалась для исходного вектора $y^{(0)} = 1000$ и исходной величины мутации $\sigma^{(0)} = 1$.

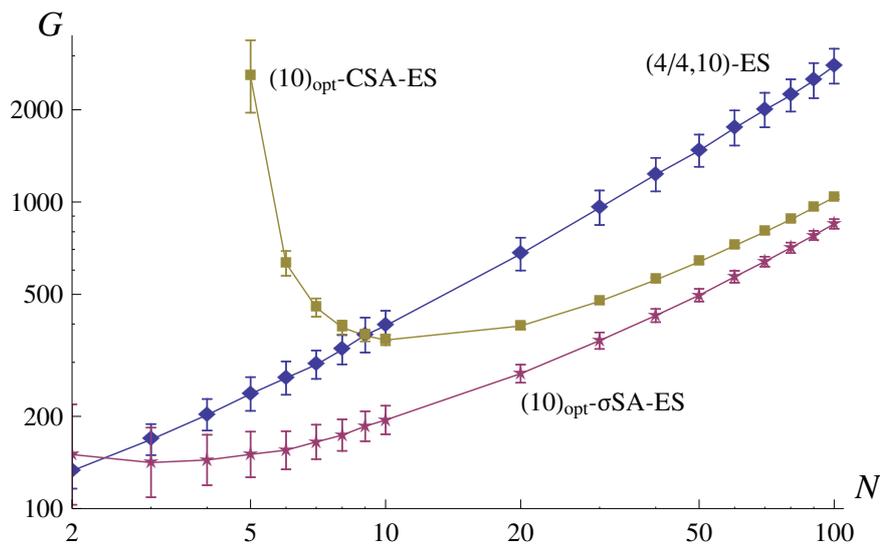
На рис. 3.31, а представлены результаты сравнения с использованием реализованного модуля ЭС и модуля построения графиков системы TALGAT. Верхняя кривая соответствует экспериментальным данным для $(4/4_I, 10)$ - σ -самоадаптационной ЭС, нижняя – для (10) - σ -самоадаптационной ЭС со значением параметра самообучения $\tau = \alpha_{\text{opt}}/\sqrt{N}$, где величина α_{opt} вычислена по формуле (2.34). Из рисунка видно, что для размерностей пространства поиска $N > 2$ $(10)_{\text{opt}}$ - σ СА-ЭС требуется меньше поколений, чтобы достичь значения целевой функции $F(y) < 10^{-10}$, чем $(4/4_I, 10)$ - σ СА-ЭС: 128 поколений против 177 для $N = 3$, 282 против 804 для $N = 20$ и 843 поколения против 3200 для $N = 100$. Это закономерный результат, так как $(\lambda)_{\text{opt}}$ - σ СА-ЭС является усовершенствованной версией стандартной $(\mu/\mu_I, \lambda)$ - σ СА-ЭС.

Сравнивая кривые на рис. 3.31, а с опубликованными в [99] результатами (рис. 3.31, б), которые были получены для аналогичных параметров ЭС с помощью математического пакета Octave, можно сделать вывод, что соответствующие графики совпадают в пределах

стандартного отклонения (изображено в виде планки погрешностей на рис. 3.31, б) и обе версии ЭС выполняют оптимизацию тестовой целевой функции (1.3) корректно.



а



б

Рис. 3.31. Тестирование реализации ЭС: а – результаты модуля ЭС; б – опубликованные результаты [99]

В дополнение к $(\mu/\mu_I, \lambda)$ - σ -самоадаптационной и $(\lambda)_{\text{opt}}$ - σ -самоадаптационной ЭС модуль ЭС реализует также две версии ЭС с адаптацией ковариационной матрицы: $(\mu/\mu_W, \lambda)$ -АКМ-ЭС и $(\mu/\mu_I, \lambda)$ -АКМ- σ СА-ЭС. Относительно этих версий ЭС выполнено эксперименталь-

ное сравнение результатов оптимизации для сферической модели целевой функции (1.3). ЭС должна найти $y_{\min} = 0$ данной тестовой функции.

В экспериментальном сравнении (рис. 3.32) подсчитывалось количество поколений G , необходимое для достижения значения целевой функции $F(y) < 10^{-10}$. Оптимизация запускалась для исходной точки поиска $y_{\text{init}} = 1$ и исходной величины мутации $\sigma^{(0)} = 1$. Для $(\mu/\mu_I, \lambda)$ -АКМ- σ СА-ЭС использовался параметр $\tau = 1/\sqrt{2N}$. Проводилось 20 независимых запусков для каждого алгоритма. На рис. 3.32 показаны средние значения количества поколений G для достижения $F(y) < 10^{-10}$.

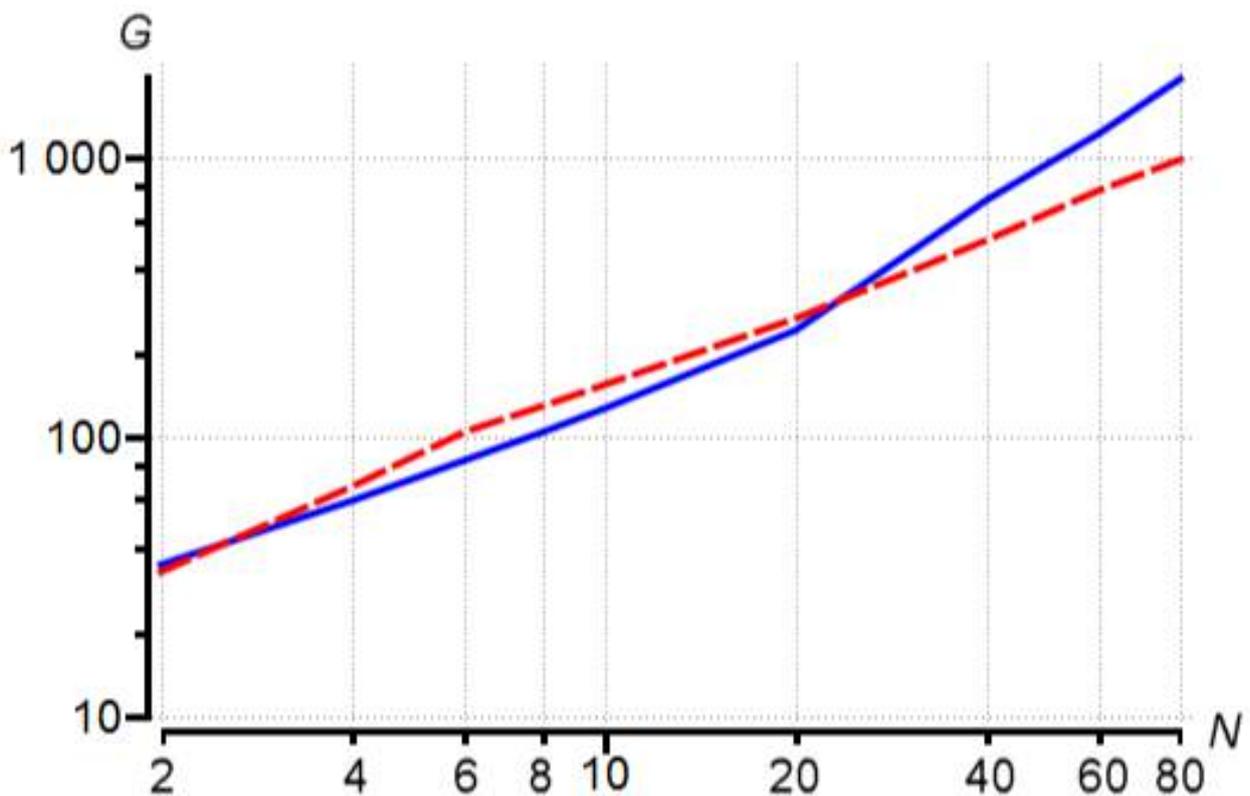


Рис. 3.32. Тестирование $(2/2_W, 8)$ -АКМ-ЭС (—) и $(2/2_I, 8)$ -АКМ- σ СА-ЭС (---)

Зависимости среднего значения количества поколений G от размерности пространства поиска N построены с помощью модуля построения графиков системы. Сплошная линия соответствует экспериментальным данным для $(2/2_W, 8)$ -АКМ-ЭС, пунктирная – для $(2/2_I, 8)$ -АКМ-σСА-ЭС.

Из рисунка видно, что для размерностей пространства поиска $N \leq 20$ разница в производительности двух версий ЭС находится в пределах погрешности усреднения. Для $N > 20$ $(2/2_I, 8)$ -АКМ-σСА-ЭС (пунктирная линия) достигает окрестности оптимума сферической функции за меньшее количество поколений по сравнению с $(2/2_W, 8)$ -АКМ-ЭС (сплошная линия). При этом обе версии ЭС выполняют оптимизацию тестовой функции, что говорит о корректности реализации $(\mu/\mu_W, \lambda)$ -АКМ-ЭС и $(\mu/\mu_I, \lambda)$ -АКМ-σСА-ЭС в модуле ЭС.

3.8. Программная реализация алгоритма неявного фильтрации

Алгоритм неявного фильтрации (НФ) изначально предложен в [106], где использовался для проектирования усилителей на арсенид-галлиевых полевых транзисторах с затвором Шоттки. В своей основе алгоритм НФ (алгоритм 3.1) – это алгоритм быстрого спуска с разностными градиентами [16]. Алгоритм НФ использует аппроксимированные градиенты в явном виде и позволяет повысить скорость оптимизации с помощью квазиньютоновских методов. Так как градиенты в алгоритме НФ вычисляются с помощью разностной аппроксимации, направление рассчитанного градиента может отличаться от направления быстрей-

шего спуска, что ведет к некорректной работе классического алгоритма быстрейшего спуска. В этом случае алгоритм НФ уменьшает значение разностного приращения и перезапускает алгоритм быстрейшего спуска.

Алгоритм 3.1 Алгоритм неявного фильтрования

функция `IMPLICITFILTERING()`

1 задать начальную точку поиска y_{init} , список разностных приращений $\{h_n\}$

2 задать параметры линейного поиска α и β

3 $k = 1; y_1 = y_{init}$

4 цикл $n = 1, \dots, \text{length}(\{h_n\})$ начать

5 $\{y_k\} = \text{SteepestDescent}(\{y_k\}, h_n, \alpha, \beta, \tau)$

6 конец

конец

функция $\{y_k\} = \text{СТЕЕПЕСТДЕСЦЕНТ}(\{y_k\}, h, \alpha, \beta, \tau)$

1 цикл $i = 1, \dots, \text{maxIterations}$ начать

2 если $\|\nabla_h F(y_k)\| \leq \tau h$ то сходимость; возврат

3 цикл $m = 0, \dots, \text{maxStepLengthReductions}$ начать

4 $d = \beta^m$

5 $y_{k+1} = y_k - d \nabla_h F(y_k)$

6 если $F(y_{k+1}) - F(y_k) < -\alpha d \|\nabla_h F(y_k)\|^2$

7 то найдено y_{k+1} ; цикл конец

8 если $m = \text{maxStepLengthReductions}$ то ошибка;

возврат

9 конец

10 $k = k + 1$

конец

За счет постоянного уменьшения значения разностного приращения алгоритм НФ увеличивает точность аппроксимации разностного градиента, что позволяет на более

поздних этапах оптимизации получить точное направление к оптимуму целевой функции. Он также выполняет неявную фильтрацию вредных шумовых составляющих целевой функции, что обеспечивает более эффективную оптимизацию реальных задач. При этом алгоритм НФ сохраняет преимущества градиентных методов оптимизации, которые для незашумленных целевых функций вычисляют точное направление градиента и сходятся к оптимуму за несколько итераций.

Алгоритм НФ реализован в составе модуля эволюционных стратегий, после загрузки которого становятся доступны команды IF_OPTS и IF_MIN. С помощью команды IF_OPTS пользователь задает вид разностной аппроксимации и начальную точку поиска. Команда IF_OPTS имеет следующие параметры.

1. Вид разностной аппроксимации градиента (0 – центральный, рекомендуемый; 1 – форвардный).
2. Начальная точка поиска.

Команда IF_MIN запускает оптимизацию и имеет следующие параметры.

1. Длина списка разностных приращений $\{h_n\}$ – *nScales* (рекомендуется *nScales* = 10).
2. Максимальное количество итераций *nIter* (по смыслу аналогично количеству поколений *nGens*).
3. Размерность пространства поиска *N* (равна количеству параметров целевой функции).
4. Имя динамической команды, которая реализует целевую функцию.

Как и в случае ЭС, правила построения целевой функции на скриптовом языке не отличаются от правил для модуля

ГА и все ранее созданные целевые функции могут использоваться без изменений.

В листинге 3.5 представлен скрипт для сравнения результатов оптимизации ГА, ЭС и НФ на примере сферической модели целевой функции, которая в первом приближении отвечает условиям работы алгоритма оптимизации в окрестности оптимума целевой функции.

В скрипте в листинге 3.5 после загрузки необходимых модулей задается целевая функция Sphere и затем запускается оптимизация с помощью ГА, ЭС и НФ с замером времени. Для $FEs = 100$ оптимизацию целевой функции с 10 параметрами потребовалось соответственно 0,171, 0,125 и 0,11 с, таким образом, по временным затратам на оптимизацию алгоритм НФ опережает ГА и ЭС. Однако более важным критерием оценки алгоритма оптимизации является его сходимость, то есть способность находить оптимум целевой функции, который для сферической модели целевой функции находится в точке 0. Информация о сходимости ГА, ЭС и НФ представлена в табл. 3.6.

Таблица 3.6

Сходимость ГА, ЭС, НФ для сферической модели целевой функции

$FEs = 100$	ГА	ЭС	НФ
$F(y^{(FEs)})$	579	110	0

Из табл. 3.6 видно, что за 100 вычислений ГА и ЭС достигли значений целевой функции 579 и 110 соответственно, в то время как алгоритм НФ сошелся к минимуму целевой функции в точке 0, что объясняется аппроксимацией НФ градиента целевой функции, которая в случае сферической модели дает точное направление наибоыстрейшего

Листинг 3.5 Скрипт для сравнения работы ГА, ЭС и НФ

```
INCLUDE "UTIL"
INCLUDE "MATRIX"
INCLUDE "ES"
SET "nGens" 10
SET "aN" 10
SET "y" CREATE_REAL_MATRIX 1 aN
CREATE_KEYWORD "Sphere"
SET "y" SET_MATRIX_ROW y 0
GA_PARAM_1 GA_PARAM_2
GA_PARAM_3 GA_PARAM_4
GA_PARAM_5 GA_PARAM_6
GA_PARAM_7 GA_PARAM_8
GA_PARAM_9 GA_PARAM_10
END_CREATE_KEYWORD SPHERE_MODEL y
ECHO TIMER GA_MIN 10 nGens 0.1 0.5 aN -10 +10 -10 +10
-10 +10 -10 +10 -10 +10 -10 +10 -10 +10 -10 +10 -10
+10 Sphere
ES_OPTS 0 1 10.
ECHO TIMER ES_MIN 3 10 nGens aN "Sphere"
IF_OPTS 0 10.
ECHO TIMER IF_MIN 4 50 aN "Sphere"
ECHO GET_BEST_IF_RESULT
ECHO GET_BEST_IF_PARAMETER 0
ECHO GET_BEST_IF_PARAMETER MINUS aN 1
```

спуска и обеспечивает сходимость за 1–2 итерации в зависимости от настроек алгоритма НФ. Таким образом, алгоритм НФ особенно эффективен для целевых функций, у которых возможно вычислить градиент и определить направление наибыстрейшего спуска. Для более сложных функций со значительными шумовыми составляющими целесообразно использовать ЭС, в то время как для дискретной оптимизации рекомендуется использовать ГА.

4. Методика оптимизации электромагнитной совместимости бортовой аппаратуры космических аппаратов

4.1. Описание методики оптимизации

Предлагаемая авторами методика оптимизации ЭМС БА КА [125] была создана для разрабатываемых на ОАО «ИСС» УЭМ БА перспективных КА и апробирована на ряде примеров оптимизации параметров: полосковых линий, математической модели резистора, многопроводной линии передачи, а также структурной оптимизации многокаскадного модального фильтра. Однако методика оптимизации ЭМС БА КА применима и для более широкого круга задач, что продемонстрировано на примере решения задачи оптимизации сосредоточенных нагрузок проводных антенн. Методика оптимизации ЭМС БА КА основана на реализованных в системе компьютерного моделирования TALGAT средствах моделирования и оптимизации (рис. 4.1).

Методика оптимизации ЭМС БА КА состоит из ряда этапов.

1. Создание ММ составной части УЭМ (описание с помощью языка скриптов, графический ввод средствами визуального клиента системы TALGAT или импорт из формата САПР).
2. Проверка корректности математической модели путем моделирования в диапазоне параметров и сравнения результатов моделирования с контрольными значениями (в качестве контрольных значений используются результаты измерений, опубликованные данные,

результаты моделирования в других программных продуктах, вычисления по аналитическим формулам).

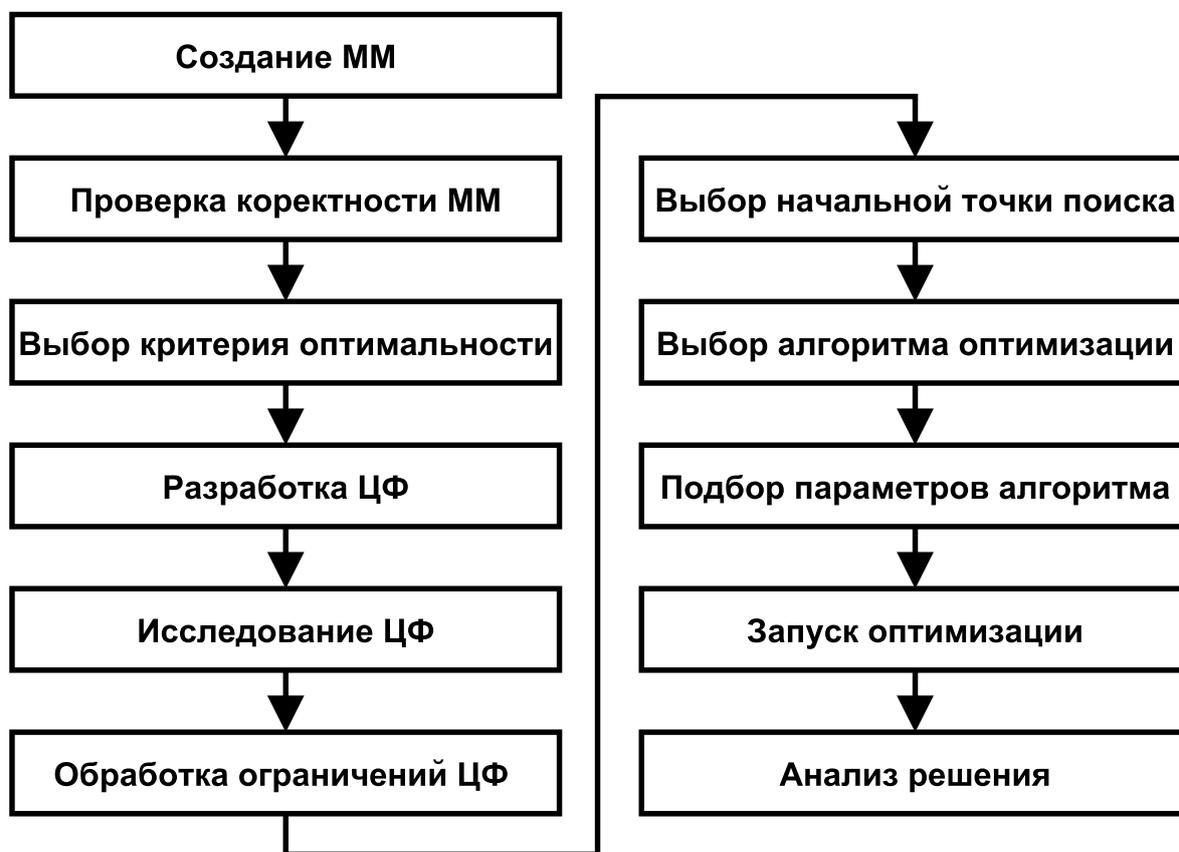


Рис. 4.1. Методика оптимизации ЭМС БА КА

3. Определение критерия оптимальности параметров математической модели (в случае нескольких критериев необходимо либо выполнить скаляризацию задачи оптимизации, либо использовать алгоритмы многокритериальной оптимизации).
4. Разработка целевой функции (ЦФ) в виде динамической команды языка скриптов системы TALGAT с повторным использованием разработанного на этапе 1 кода математической модели (в качестве результата целевая функция должна возвращать вещественное число).

5. Исследование характеристик и оценка сложности целевой функции путем сбора статистических данных о ее свойствах, построения одномерных или двумерных графиков зависимости значений целевой функции от одного или двух параметров при фиксированных остальных параметрах (фиксированные значения параметров выбираются либо на основании экспертных рекомендаций, либо из середины диапазона допустимых значений; при необходимости, в частности в случае большой сложности рельефа целевой функции, большого количества локальных экстремумов, используется несколько наборов фиксированных значений).
6. Обработка ограничений целевой функции. В задачах ЭМС в большинстве случаев параметры имеют двусторонние ограничения, которые могут обрабатываться с помощью штрафных функций или функций-декодеров (если границы значений различных параметров отличаются на несколько порядков, для алгоритмов оптимизации с адаптивным шагом может применяться преобразование значений параметров целевой функции в диапазон $[0, 1]$ с целью ускорения сходимости на первоначальном этапе оптимизации; в этом случае целевая функция выполняет обратное преобразование параметров из диапазона $[0, 1]$ в исходный диапазон).
7. Выбор начальной точки поиска: использование генераторов случайных чисел, выборки латинского гиперкуба, рекомендаций экспертов, результатов предыдущих запусков алгоритма оптимизации, полученных на этапе 5 сведений о рельефе целевой функции (для таких популяционных алгоритмов оптимизации, как ЭС и ГА, возможен выбор нескольких начальных точек поиска).

8. Выбор алгоритма оптимизации на основании характеристик целевой функции, результатов запусков алгоритмов оптимизации для небольшого количества итераций или упрощенных версий целевой функции (упрощение целевой функции выполняется путем фиксации параметров, снижения точности математической модели, использования менее точных алгоритмов моделирования и решения СЛАУ, уменьшения количества рассматриваемых точек в случае применения в целевой функции анализа во временном или частотном диапазоне).
9. Подбор параметров алгоритма оптимизации: использование рекомендованных в литературе значений, выбор значений на основании эмпирических данных, вычисление по аналитическим формулам, автоматический выбор параметров, реализуемый за счет использования эволюционных алгоритмов.
10. Запуск алгоритма оптимизации. Если используемый компьютер имеет многоядерную и/или многопроцессорную архитектуру, то система TALGAT позволяет сократить время оптимизации за счет вычисления целевой функции в несколько потоков.
11. Анализ полученного решения задачи оптимизации: при необходимости, построение графиков изменения значений целевой функции в процессе оптимизации для оценки сходимости алгоритма, статистический анализ чувствительности найденного оптимального решения к отклонениям реальных параметров от расчетных, например с помощью метода Монте-Карло.

Так как отдельные этапы тесно взаимосвязаны, допускается их выполнение в другой последовательности, а также возврат к предыдущим этапам в случае выявления на каком-либо этапе новых сведений о характеристиках целевой функции, поведении алгоритма оптимизации и т.п.

4.2. Оптимизация параметров полосковых линий

Рассмотрим применение предложенной методики для оптимизации параметров полосковой линии по критерию минимизации коэффициентов связи в межсоединениях печатной платы с двухслойным диэлектриком [107]. В таких платах сигнал часто распространяется, например, от управляющего вывода вдоль обращенной полосковой линии (ОПЛ, рис. 4.2, а) и затем через сквозное отверстие вдоль подвешенной полосковой линии (ППЛ, рис. 4.2, б) к приемному порту [108].

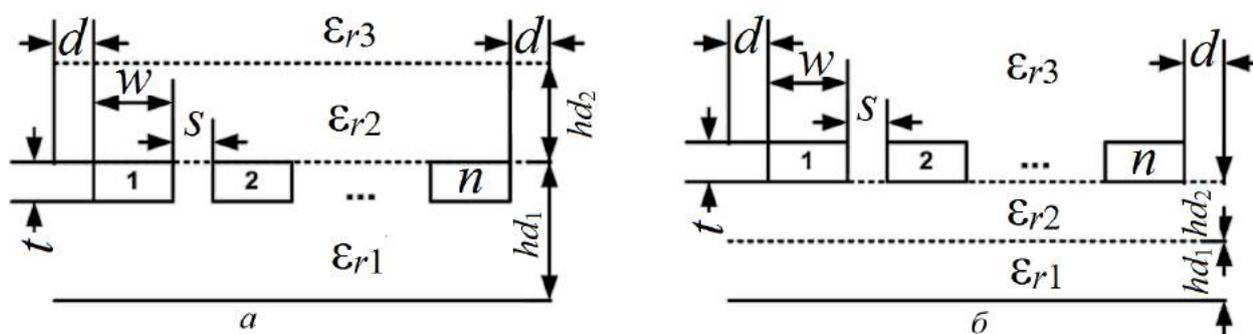


Рис. 4.2. Обращенные (а) и подвешенные (б) полосковые линии

Математическая модель. Входными параметрами ММ представленных на рис. 4.2 ОПЛ и ППЛ являются количество линий n , значения относительной диэлектрической проницаемости ϵ_{r1} , ϵ_{r2} , ϵ_{r3} , геометрические параметры d , s , w , t , h_{d1} и h_{d2} . В соответствии с этапом 1 методики

оптимизации ЭМС БА КА математическая модель ОПЛ и ППЛ записывается с помощью языка скриптов системы TALGAT. Для этого используются команды модуля двумерного электростатического анализа MOM2D, модуля команд общего назначения UTIL и модуля работы с матрицами MATRIX. Полученный в результате скрипт для моделирования ППЛ представлен в листинге 4.1. Для краткости опущен код динамической команды `add_snd_up` (для создания дополнительных участков ППЛ с ростом числа проводников). Скрипт для моделирования ОПЛ аналогичен скрипту в листинге 4.1.

Для уменьшения количества параметров ММ выбраны следующие фиксированные значения параметров (см. рис. 4.2): $n = 2$; $\epsilon_{r1} = 2$; $\epsilon_{r2} = 5$; $\epsilon_{r3} = 1$; $d=s=w$; $t=0,1w$; $h_{d1} = 0,5w$. Таким образом, количество параметров ММ сокращено до двух: ширина полоски w и высота слоя диэлектрика h_{d2} .

В соответствии с этапом 2 методики оптимизации ЭМС БА КА корректность ММ ОПЛ и ППЛ проверена путем моделирования в диапазоне параметров и сравнения результатов моделирования с данными, опубликованными в [108].

Целевая функция. Этап 3 методики оптимизации ЭМС БА КА заключается в выборе критерия оптимальности параметров ММ. В качестве критерия оптимальности параметров представленной выше ММ выбран критерий минимальной разницы емкостной и индуктивной связи $K_C - K_L$ для ППЛ и ОПЛ. Если функции $K_C - K_L$ для ОПЛ и ППЛ равны по абсолютному значению и противоположны по знаку, наблюдается эффект полной компенсации дальней перекрестной помехи ППЛ перекрестной помехой ОПЛ [108].

Листинг 4.1 . Математическая модель ППЛ на языке скрип-

тов системы TALGAT

```
INCLUDE "MOM2D"
```

```
INCLUDE "UTIL"
```

```
INCLUDE "MATRIX"
```

```
SET _INFINITE_GROUND 1
```

```
CYCLE num_of_lines add_cnd_up // создание
```

```
num_of_lines участков ППЛ
```

```
DIELECTRIC // создание диэлектрика
```

```
    SET _SUBINTERVALS si_diel_btw
```

```
    SET _ER_PLUS er2
```

```
    SET _ER_MINUS er3
```

```
    LINE MINUS x_begin s hd2 PLUS MINUS x_begin s d  
hd2
```

```
    SET _SUBINTERVALS si_diel
```

```
    SET _ER_PLUS er1
```

```
    SET _ER_MINUS er2
```

```
    LINE 0. hd1 PLUS MINUS x_begin s d hd1
```

```
SET "conf" GET_CONFIGURATION_2D // сохранение  
конфигурации
```

```
SET "smn" SMN_C conf // вычисление матриц погонных  
параметров ППЛ
```

```
SET "cm" CALCULATE_C smn conf
```

```
SET "smn_l0" SMN_L0 conf
```

```
SET "cm_l0" CALCULATE_L0 smn_l0 conf
```

Обозначим разницу функций $K_C - K_L$ для ОПЛ и ППЛ как $\Delta K_C K_L$, тогда $\Delta K_C K_L = 0$ при оптимальном значении параметров w и h_{d2} .

Чтобы выполнить этап 4 методики оптимизации ЭМС ВА КА, разработанные скрипты для моделирования ОПЛ и ППЛ были объединены в одну динамическую команду `qf_unconstrained` (листинг 4.2), которая реализует вычисление целевой функции и возвращает вещественное число – значение $\Delta K_C K_L$.

Листинг 4.2 . Целевая функция `qf_unconstrained`

```
CREATE_KEYWORD "qf_unconstrained" // вычисление
матриц погонных параметров ППЛ (см. листинг 4.1)
SET "kc" DIV MINUS 0. GET_VALUE cm C12_row
C12_col GET_VALUE cm C11_row C11_col
SET "kl" DIV GET_VALUE cm_l0 C12_row C12_col
GET_VALUE cm_l0 C11_row C11_col
SET "kc_up" MINUS kc kl // ( $K_C - K_L$ ) для ППЛ
// вычисление матриц погонных параметров ОПЛ
SET "kc" DIV MINUS 0. GET_VALUE cm C12_row
C12_col GET_VALUE cm C11_row C11_col
SET "kl" DIV GET_VALUE cm_l0 C12_row C12_col
GET_VALUE cm_l0 C11_row C11_col
SET "kc_down" MINUS kc kl // ( $K_C - K_L$ ) для ОПЛ
END_CREATE_KEYWORD MINUS ABS kc_up
ABS kc_down // расчет  $\Delta K_C K_L$ 
```

Для оценки характеристик целевой функции в рамках выполнения этапа 5 построены двумерные контурные графики зависимости значений $\Delta K_C K_L$ целевой функции от параметров w и h_{d2} для диапазонов $w = h_{d2} = 0; 0,01; \dots; 1$ мм (рис. 4.3). Изолинии на рис. 4.3

проведены по точкам равных значений $\Delta K_C K_L$. Стрелкой с надписью min указан глобальный минимум целевой функции.

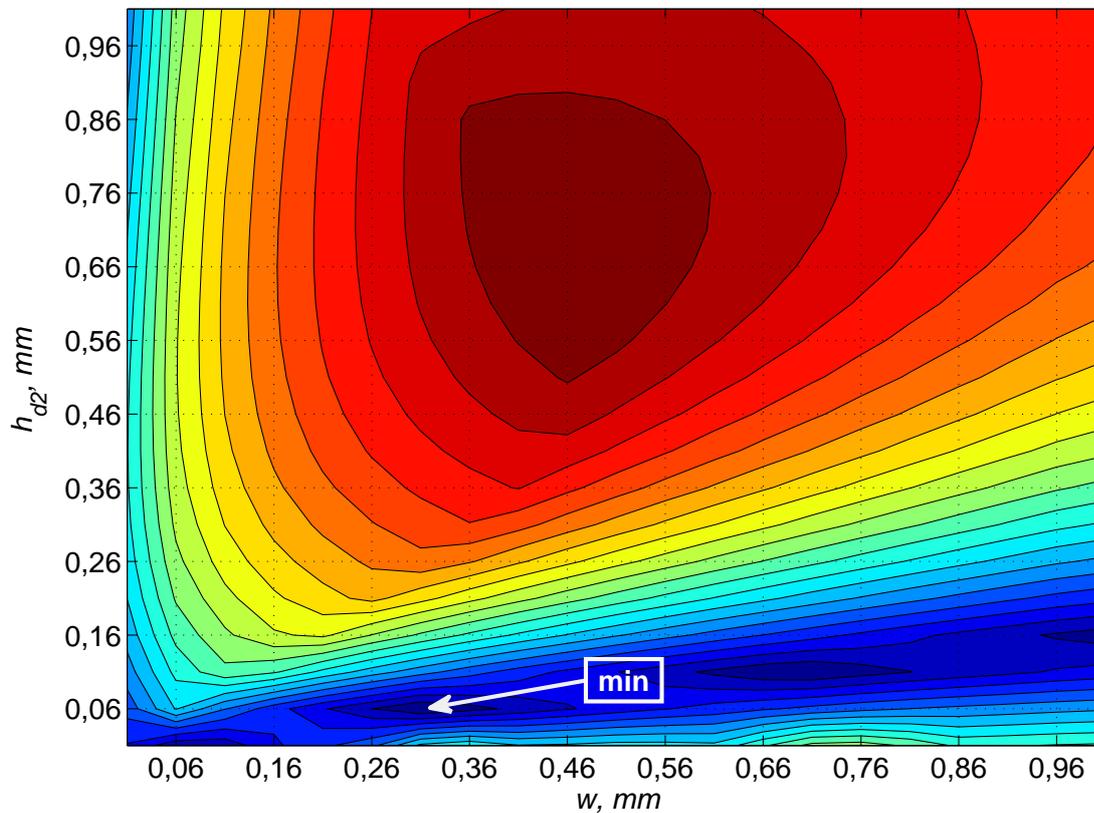


Рис. 4.3. Зависимость значений $\Delta K_C K_L$ целевой функции от параметров w и h_{d2}

Из рис. 4.3 видно, что целевая функция для рассмотренного диапазона значений параметров помимо глобального минимума (показан белой стрелкой с надписью min) имеет 3 локальных минимума в области $h_{d2} < 0,25$ мм (три изолированных темных участка в нижней части рисунка). Однако рельеф целевой функции является гладким, отсутствуют разрывы и неопределенности и в целом целевая функция не представляет сложности для оптимизации.

Рассматриваемая целевая функция имеет ограничения параметров. Диапазон допустимых значений параметров соответствует показанному на рис. 4.3 диапазону $w = h_{d2} = 0; 0,01; \dots; 1$ мм. В качестве метода обработки ограничений целевой функции выбран метод штрафных функций: при выходе значения параметра x за границу допустимого диапазона x_{limit} вместо значения целевой функции подставляется разница $|x - x_{\text{limit}}|$, умноженная на коэффициент штрафной функции. Обработка ограничений целевой функции также реализована на языке скриптов, коэффициент штрафной функции выбран равным 10 с учетом того, что значения $\Delta K_C K_L$ на рис. 4.3 не превышают единицы. В качестве начальных значений параметров взята величина 0,5 мм, близкая к середине допустимого диапазона параметров.

Сравнение алгоритмов оптимизации. На этапе 8 методики оптимизации ЭМС БА КА выбирается алгоритм оптимизации на основании характеристик целевой функции. Учитывая, что целевая функция имеет размерность $N = 2$ и гладкий рельеф, для поиска решения целесообразно использовать методы оптимизации на основе аппроксимации градиента, например алгоритм неявного фильтрования.

Для сравнения оптимизация целевой функции выполнялась с помощью алгоритма НФ, ГА, $(\mu/\mu_I, \lambda)$ - σ СА-ЭС, $(\lambda)_{\text{opt}}$ - σ СА-ЭС и $(\mu/\mu_W, \lambda)$ -АКМ-ЭС (рис. 4.4, б). Параметры алгоритмов:

- 1) НФ – $nScales = 20$, $\alpha = 1$, $\beta = 0,1$, $\tau = 1$, $maxStepLengthReductions = 10$, $maxIterations = 1000$;
- 2) ГА – коэффициент мутации 0,06, коэффициент кроссовера 0,6, размер популяции 10;

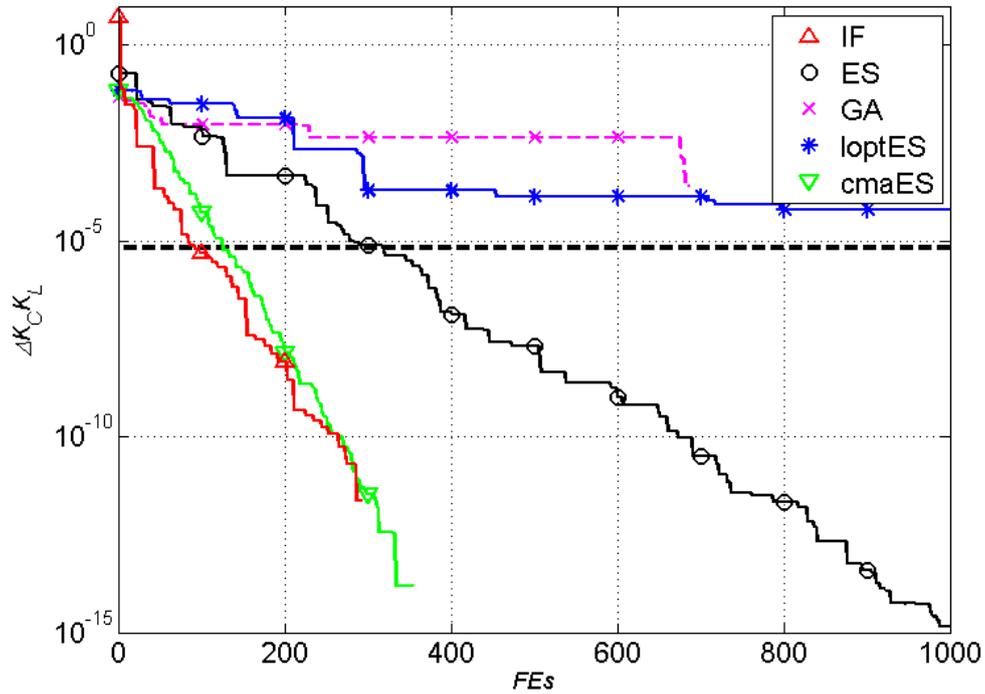
- 3) $(\mu/\mu_I, \lambda)$ - σ СА-ЭС – $\mu = 3, \lambda = 10, \tau = 1/\sqrt{N}, \sigma^{(0)} = 0,3$;
 4) $(\lambda)_{\text{opt}}$ - σ СА-ЭС – $\mu = 3, \lambda = 10, \tau = 8,6/\sqrt{N}$ (расчет по формуле (2.34)), $\sigma^{(0)} = 0,3$;
 5) $(\mu/\mu_W, \lambda)$ -АКМ-ЭС – $\mu = 3, \lambda = 6, \sigma^{(0)} = 0,3$.

Параметры алгоритма НФ и ГА выбраны на основании эмпирических данных, для параметров $(\mu/\mu_I, \lambda)$ - σ СА-ЭС и $(\mu/\mu_W, \lambda)$ -АКМ-ЭС взяты рекомендованные в литературе значения. Помимо случая с двумя параметрами целевой функции, на рис. 4.4, *a* представлен частный случай целевой функции с одним параметром h_{d2}/w , в которой значение параметра $w = 1$ мм было зафиксировано.

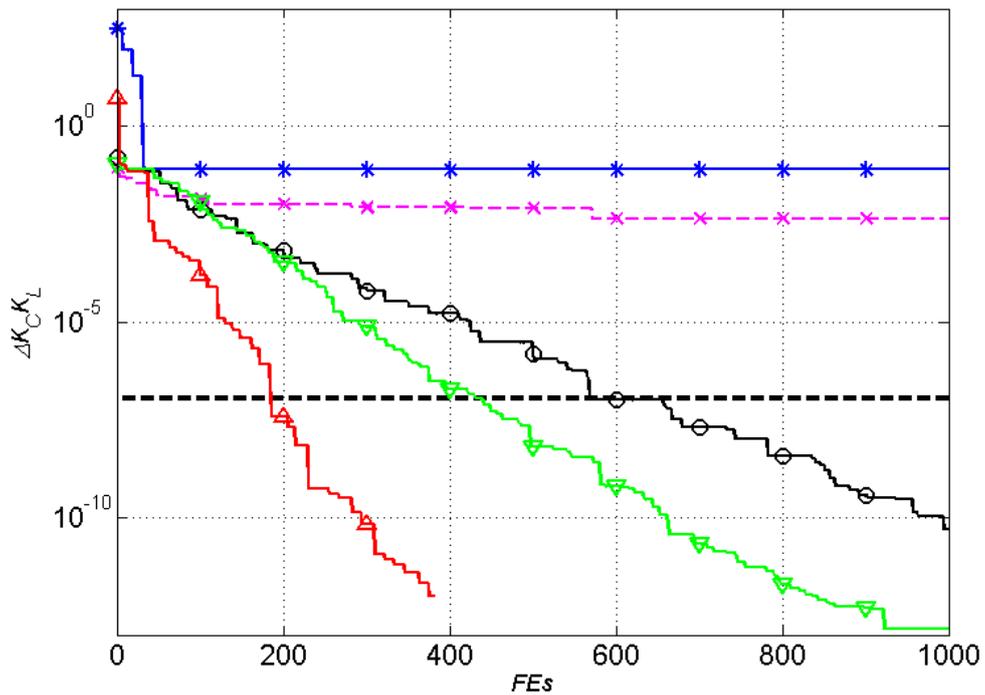
Экспериментальные данные для рис. 4.4 собраны следующим образом. После запуска алгоритма при первом вычислении⁴ целевой функции значение $\Delta K_C K_L$ сохранялось в качестве минимального Δ_{\min} . Затем при каждом вычислении целевой функции полученное значение $\Delta K_C K_L$ сравнивалось с Δ_{\min} и сохранялось как новое значение Δ_{\min} в случае $\Delta K_C K_L < \Delta_{\min}$. В результате значения $\Delta K_C K_L$ на рис. 4.4 изменяются только в случае, если алгоритм оптимизации получает меньшее значение целевой функции. На отрезках, где алгоритм не находит меньших значений $\Delta K_C K_L$, линия остается горизонтальной.

На рис. 4.4 пунктирной линией показано минимальное значение целевой функции, найденное путем перебора при вычислении значений целевой функции в 10^4 точек. Для целевой функции с одним параметром h_{d2}/w (см. рис. 4.4, *a*) алгоритм НФ и АКМ-ЭС показывают лучшую производительность и достигают меньшего значения целевой

⁴ В случае стохастических алгоритмов первое значение $\Delta K_C K_L$ вычислялось для полученного с помощью мутации вектора параметров, поэтому для ГА и ЭС начальная точка графика на рис. 4.4 имеет случайное значение.



a



б

Рис. 4.4. Усредненная по 10 запускам зависимость $\Delta K_C K_L$ от количества вычислений целевой функции FEs для разного числа параметров: *a* – одного (h_{d2}/w); *б* – двух (w и h_{d2}). Δ – НФ; ∇ – $(\mu/\mu_W, \lambda)$ -АКМ-ЭС; \circ – $(\mu/\mu_I, \lambda)$ - σ СА-ЭС; \times – ГА; $*$ – $(\lambda)_{opt}$ - σ СА-ЭС

функции, чем перебор для $FEs = 10^4$ вычислений целевой функции, уже при $FEs = 150$, $(3/3_I, 10)$ -σСА-ЭС требуется для этого $FEs = 300$. Все три алгоритма достигают значения $\Delta K_C K_L < 10^{-10}$ в рамках бюджета из 1000 вычислений целевой функции. ГА и $(10)_{opt}$ -σСА-ЭС показывают плохую сходимость для данной целевой функции, так как они предназначены для работы с целевыми функциями большей размерности.

Для целевой функции с двумя параметрами (w и h_{d2} , см. рис. 4.4, б) лучшую производительность показывает алгоритм НФ, который достигает значения $\Delta K_C K_L < 10^{-10}$ за $FEs = 200$ вычислений целевой функции. За 45 вычислений целевой функции с помощью алгоритма НФ получена разница емкостной и индуктивной связей для ППЛ и ОПЛ менее 10^{-3} , что вполне достаточно для практического применения. На втором и третьем месте по производительности оказались АКМ-ЭС и $(3/3_I, 10)$ -σСА-ЭС соответственно. ГА и $(10)_{opt}$ -σСА-ЭС продемонстрировали плохую сходимость аналогично случаю целевой функции с одной переменной. Таким образом, алгоритм НФ показал лучшие результаты для рассматриваемой целевой функции и был выбран для оптимизации параметров полосковой линии.

Подбор параметров алгоритма неявного фильтрования. На этапе 9 методики оптимизации ЭМС БА КА выполняется подбор параметров алгоритма оптимизации. Для алгоритма НФ применен автоматический выбор параметров, реализованный за счет поиска их оптимальных значений с помощью $(\mu/\mu_W, \lambda)$ -АКМ-ЭС. В качестве критерия оптимальности взят критерий достижения НФ минимального значения

целевой функции `qf_unconstrained` (см. листинг 4.2) в пределах заданного бюджета вычислений целевой функции $FES = 1000$.

В качестве начального набора параметров приняты рекомендованные в [16] значения: $\alpha = 10^{-4}$, $\beta = 0,5$, $\tau = 0,01$, $maxStepLengthReductions = 3$. Параметры $nScales$ и $maxIterations$ определяют максимальное количество итераций и требуемую точность алгоритма НФ, для этих параметров выбраны фиксированные значения $nScales = 8$ и $maxIterations = 1000$. Для вызова ЭС на языке скриптов TALGAT написана целевая функция `qf_IFparam` (листинг 4.3), которая записывает набор параметров алгоритма НФ с помощью команды `IF_OPTS`, запускает оптимизацию алгоритма НФ для целевой функции `qf` (версия функции `qf_unconstrained` из листинга 4.2 с обработкой граничных условий) и возвращает найденное алгоритмом НФ минимальное значение `qf` (рис. 4.5).

После запуска ЭС создает популяцию из λ потомков ($\lambda = 3$ на рис. 4.5), каждый из которых содержит вектор с набором параметров алгоритма НФ $p_1, p_2, \dots, p_\lambda$. Затем с каждым набором параметров запускается алгоритм НФ, который оптимизирует целевую функцию `qf` (непосредственно задача оптимизации параметров полосковой линии). Для найденных в результате выполнения алгоритма НФ точек $y_1, y_2, \dots, y_\lambda$ вычисляются минимальные значения $F(y_1), F(y_2), \dots, F(y_\lambda)$ целевой функции `qf`, которые затем возвращаются ЭС в качестве значений целевой функции `qf_IFparam`. ЭС выполняет сортировку потомков по убыванию значений $F(y_1), F(y_2), \dots, F(y_\lambda)$ и генерирует следующее поколение с новыми наборами параметров алгоритма НФ.

Листинг 4.3 . Целевая функция `qf_IFparam`

```
CREATE_KEYWORD "qf_IFparam"  
IF_OPTS "alpha" GA_PARAM_1  
IF_OPTS "tau" GA_PARAM_2  
IF_OPTS "beta" GA_PARAM_3  
IF_OPTS "maxStepLengthReductions" GA_PARAM_4  
IF_MIN 8 1000 1 qf  
END_CREATE_KEYWORD GET_BEST_IF_RESULT
```

Принцип работы ЭС с целевой функцией `qf_IFparam` показан на рис. 4.5.

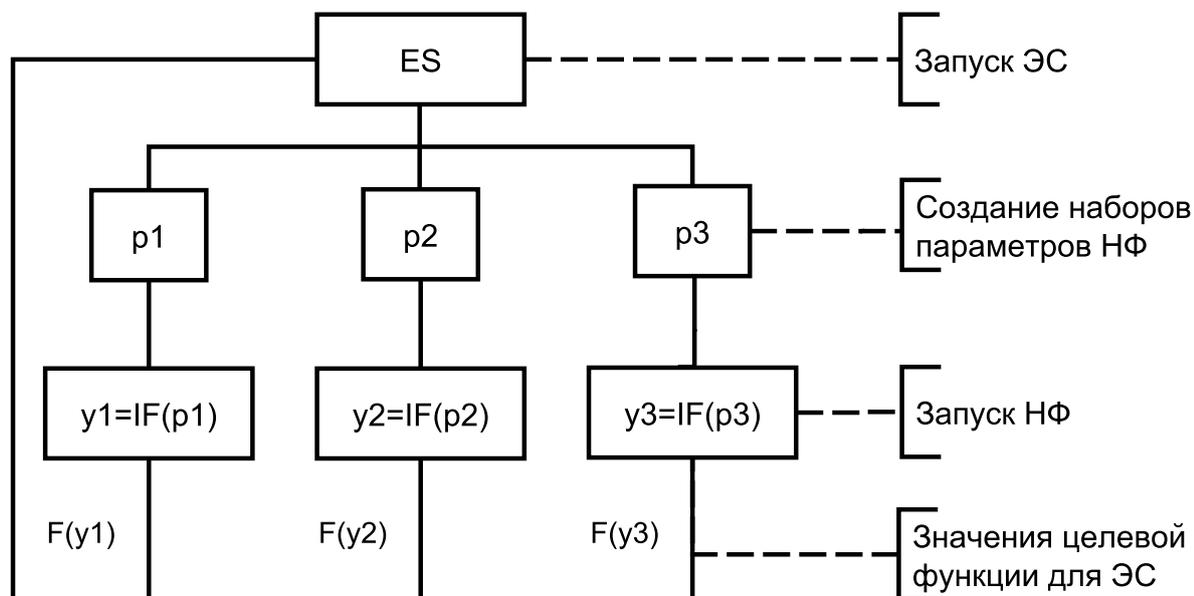


Рис. 4.5. Принцип работы эволюционной стратегии с целевой функцией `qf_IFparam`

Зависимость минимального значения $F(y_{\min})$ «вложенной» целевой функции `qf`, которая оптимизируется алгоритмом НФ от количества вычислений «основной» целевой функции `qf_IFparam`, которая оптимизируется ЭС, демонстрирует рис. 4.6.

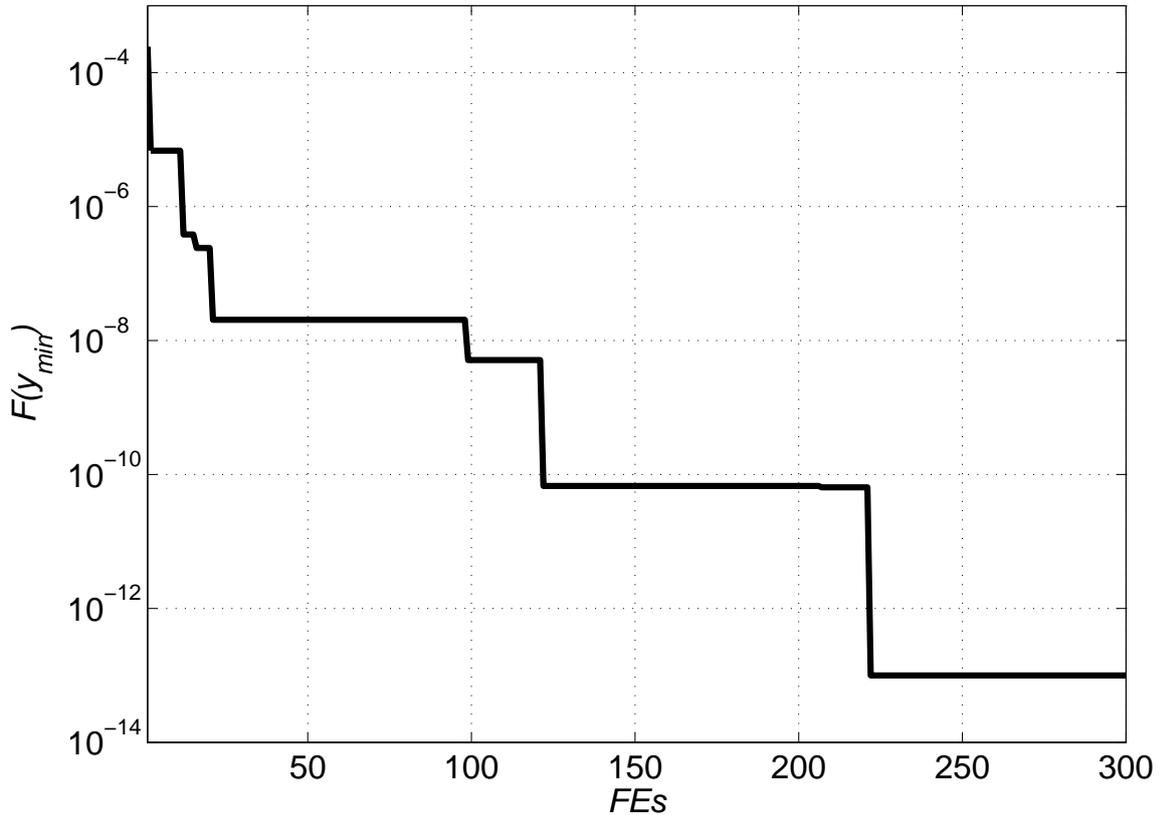


Рис. 4.6. Зависимость значения $F(y_{\min})$ целевой функции алгоритма НФ от количества вычислений целевой функции ЭС

Так как для начального набора параметров алгоритма НФ целевая функция `qf_IГрарам` из листинга 4.3 возвратила значение $2,4 \cdot 10^{-4}$, то график на рис. 4.6 начинается из этой точки. Затем ЭС за 100 вычислений целевой функции (что соответствует 100 запускам алгоритма НФ) находит такой набор параметров алгоритма НФ, который позволяет ему достичь значения $F(y_{\min}) < 10^{-8}$. Достаточно 220 вычислений целевой функции ЭС для нахождения оптимального набора параметров $\alpha = 0,6$, $\beta = 0,01$, $\tau = 0,2$, $maxStepLengthReductions = 5$, который обеспечивает достижение алгоритмом НФ значения целевой функции $F(y_{\min}) = 9,96 \cdot 10^{-14}$.

Таким образом, с помощью $(\mu/\mu_W, \lambda)$ -АКМ-ЭС автоматически выбран набор параметров алгоритма НФ, который при бюджете 1000 вычислений целевой функции обеспечивает сходимость алгоритма НФ к $F(y_{\min}) = 9,96 \cdot 10^{-14}$. В экспериментах (см. рис. 4.4, б) алгоритм НФ при том же бюджете сошелся к $F(y_{\min}) = 9,96 \cdot 10^{-13}$, то есть автоматически выбранные параметры обеспечили более высокую точность по сравнению с подобранными на основании эмпирических данных.

4.3. Оптимизация параметров проводных структур

Рассмотрим оптимизацию параметров широкодиапазонной быстроразворачиваемой модифицированной вибраторной антенны (рис. 4.7) с сосредоточенными нагрузками в виде 10 параллельных RLC -контуров [109–111]. Параметры нагрузок в плечах антенны приведены в табл. 4.1. В [15] с помощью системы TALGAT выполнено имитационное моделирование антенны, представленное на рис. 4.7, в диапазоне частот от 5 до 30 МГц. Было достигнуто хорошее совпадение КСВ с данными, полученными в системе 4NEC2 [112] (среднеквадратичное отклонение 1,62), что говорит о корректности программной реализации модуля MOMW, и в частности методики Альтмана [113], для учета сосредоточенных нагрузок.

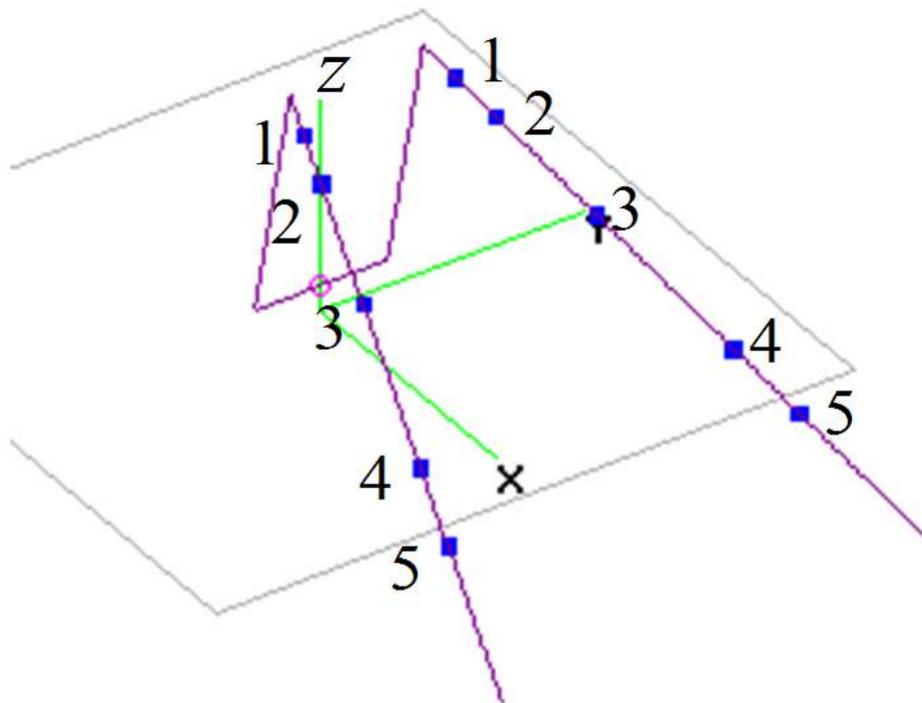


Рис. 4.7. Геометрическая модель широкодиапазонной быстроразворачиваемой модифицированной вибраторной антенны

Таблица 4.1

Параметры нагрузок широкодиапазонной быстроразворачиваемой модифицированной вибраторной антенны

Номер нагрузки	Номер сегмента	R , Ом	L , мкГн	C , пФ
1	93	810	7,85	5
2	85	810	19,80	5
3	65	810	62,54	5
4	38	810	121,8	10
5	25	810	150,7	10

Математическая модель. Входными параметрами ММ оптимизируемой антенны являются геометрические параметры структуры, расположение и параметры нагрузок. С помощью команд модуля электродинамического анализа трехмерных проводных структур MOMW

математическая модель антенны записана на языке скриптов системы TALGAT. Для уменьшения количества параметров ММ геометрические параметры структуры антенны, расположение нагрузок и значения R зафиксированы в соответствии с рис. 4.7 и табл. 4.1. В результате получена ММ с 20 параметрами: L_1, L_2, \dots, L_{10} и C_1, C_2, \dots, C_{10} . Учитывая симметричность структуры антенны, значения параметров L_1, L_6, L_2, L_7 и т.д. должны быть попарно равны, что является дополнительным критерием оценки корректности результатов оптимизации. То же относится и к параметрам C_1, C_2, \dots, C_{10} .

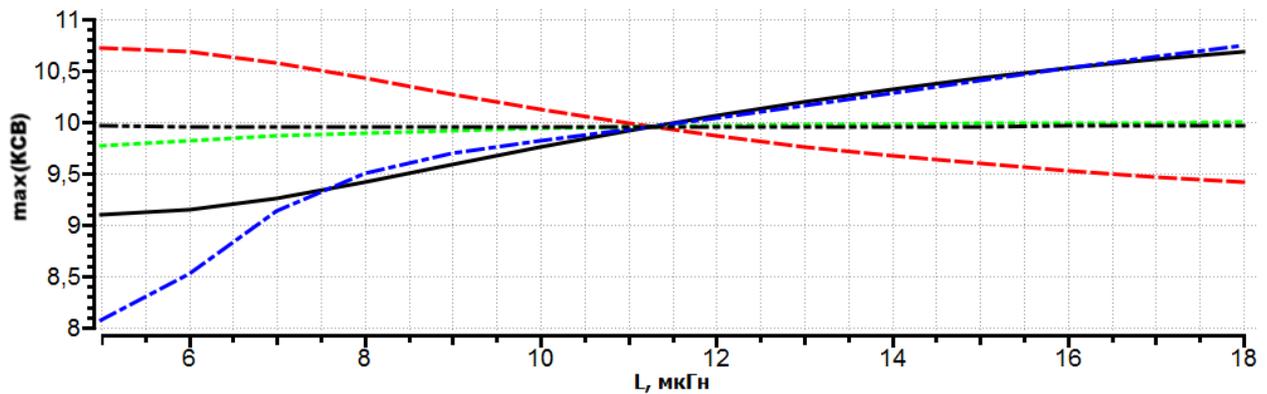
В соответствии с этапом 2 методики оптимизации ЭМС БА КА корректность ММ проверена путем вычисления КСВ в диапазоне частот от 5 до 30 МГц и сравнения результатов с данными из [15]. Результаты совпали.

Целевая функция. В качестве критерия оптимальности параметров антенны выбран критерий максимального снижения значения КСВ одновременно в трех частотных точках: 10, 20 и 30 МГц. На основе кода ММ антенны реализована целевая функция, которая вычисляет для частот 10, 20 и 30 МГц значения $\max(\text{КСВ})$, выбирает из них наибольшее и возвращает его в качестве значения целевой функции.

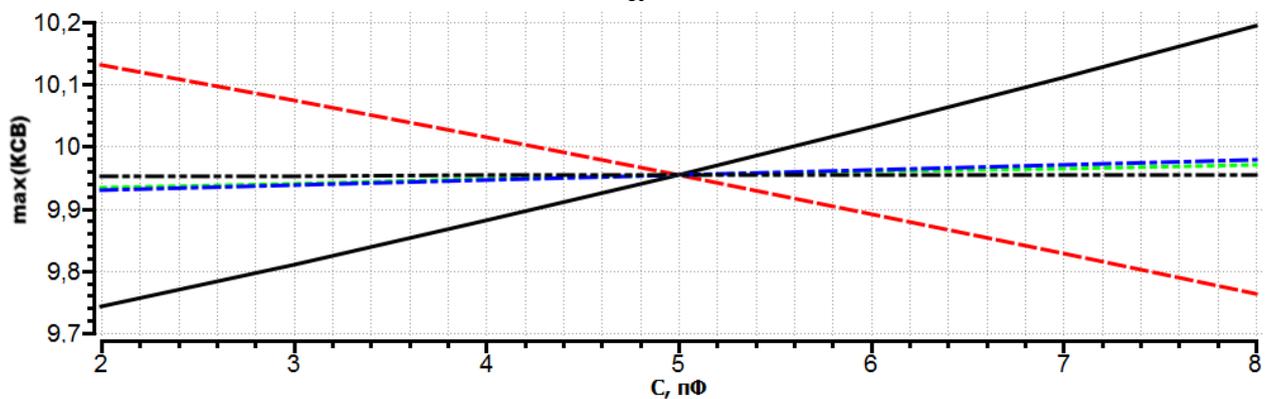
Для параметров целевой функции выбраны диапазоны 4,5–18 мкГн для L_1, L_2, \dots, L_{10} и 2–8 пФ для C_1, C_2, \dots, C_{10} . В качестве начальных значений параметров взяты значения L и C из середины допустимого диапазона параметров – 11,25 мкГн и 5 пФ.

Для оценки характеристик целевой функции и проверки корректности ее реализации проведены следующие вычислительные эксперименты: всем параметрам целе-

вой функции присваивались начальные значения, затем один из 20 параметров варьировался в допустимом диапазоне. На рис. 4.8, а показана зависимость $\max(\text{КСВ})$ от L_1, L_2, \dots, L_5 , а на рис. 4.8, б – от C_1, C_2, \dots, C_5 . Так как структура антенны и размещение нагрузок симметричны, зависимости от L_6, L_7, \dots, L_{10} и C_6, C_7, \dots, C_{10} совпадают с рис. 4.8.



а



б

Рис. 4.8. Зависимость $\max(\text{КСВ})$ от значений L_1, L_2, \dots, L_5 (а) и C_1, C_2, \dots, C_5 (б): — L_1, C_1 , - - L_2, C_2 , ··· L_3, C_3 , - · - L_4, C_4 , - ··· L_5, C_5

Из рис. 4.8, а видно, что зависимость $\max(\text{КСВ})$ от значений L_1, L_2, \dots, L_5 является нелинейной, тогда как зависимость от C_1, C_2, \dots, C_5 (см. рис. 4.8, б) в диапазоне 2–8 пФ близка к линейной.

Особенностью рассматриваемой целевой функции является значительное отличие значений параметров L_1, L_2, \dots, L_{10} , имеющих порядок 10^{-6} , от значений параметров C_1, C_2, \dots, C_{10} , которые имеют порядок 10^{-12} . Если данная особенность не представляет проблемы для ГА, в котором значения параметров кодируются в виде заданного количества битов, то для $(\mu/\mu_W, \lambda)$ -АКМ-ЭС необходимы затраты на вычисление целевой функции для адаптации к такому большому различию между значениями отдельных параметров. Алгоритмы НФ, $(\mu/\mu_I, \lambda)$ -σСА-ЭС и $(\lambda)_{\text{opt}}$ -σСА-ЭС не имеют средств для адаптации к таким особенностям целевой функции и будут показывать плохую сходимость. Для решения данной проблемы на языке скриптов реализовано масштабирование параметров L_1, L_2, \dots, L_{10} и C_1, C_2, \dots, C_{10} путем умножения их значений на 10^6 и 10^{12} соответственно. Внутри целевой функции непосредственно перед вызовом кода моделирования антенны выполняется обратное преобразование параметров.

Автор алгоритма НФ [16] при решении практических задач рекомендует преобразовывать значения параметров целевой функции в диапазон $[0; 1]$. Данный метод был также реализован на языке скриптов в виде функции-декодера (листинг 4.4) для обратного преобразования значений параметров из диапазона $[0; 1]$ в исходный диапазон.

Для обработки ограничений использован метод штрафных функций: при выходе значения параметра x за границу допустимого диапазона x_{limit} вместо значения КСВ подставлялось число 20, к которому прибавлялась разница $|x - x_{\text{limit}}|$, умноженная на коэффициент штрафной функции 10.

Листинг 4.4 . Функция-декодер для обратного преобразования параметров

SET "L1" PLUS 4.5e-6 MUL L1 13.5e-6 // аналогичная операция для L2–L9

SET "L10" PLUS 4.5e-6 MUL L10 13.5e-6

SET "C1" PLUS 2.e-12 MUL C1 6.e-12 // аналогичная операция для C2–C9

SET "C10" PLUS 2.e-12 MUL C10 6.e-12

Сравнение алгоритмов оптимизации. Результаты оптимизации целевой функции с помощью алгоритма НФ, ГА, $(\mu/\mu_I, \lambda)$ - σ СА-ЭС, $(\lambda)_{\text{opt}}$ - σ СА-ЭС и $(\mu/\mu_W, \lambda)$ -АКМ-ЭС приведены на рис. 4.9. Экспериментальные данные получены тем же способом, что и для рис. 4.4. Использовались те же параметры алгоритмов оптимизации, за исключением $(\mu/\mu_W, \lambda)$ -АКМ-ЭС, для которой было задано $\mu = 6$, $\lambda = 12$.

На рис. 4.9, а приведены результаты экспериментов для целевой функции с масштабированием параметров. Лучшую производительность показывает ГА, который достигает $\max(\text{КСВ})=7,36$ за $FES=100$ вычислений целевой функции и $\max(\text{КСВ})=6,36$ за $FES=900$. Алгоритм $(\mu/\mu_W, \lambda)$ -АКМ-ЭС имеет близкую к $(\mu/\mu_I, \lambda)$ - σ СА-ЭС производительность и достигает $\max(\text{КСВ})=6,48$ за $FES = 386$. Алгоритм $(\lambda)_{\text{opt}}$ - σ СА-ЭС показывает сходимость для $FES = 100$ и достигает $\max(\text{КСВ})=7,27$ за $FES = 564$. Алгоритм НФ для данной целевой функции не сходится. Результаты экспериментов для целевой функции с функцией-декодером показаны на рис. 4.9, б. За счет преобразования значений параметров целевой функции в диапазон $[0; 1]$ алгоритм НФ показывает лучшую производительность для $FES < 100$ и сходится к $\max(\text{КСВ})=7,08$.

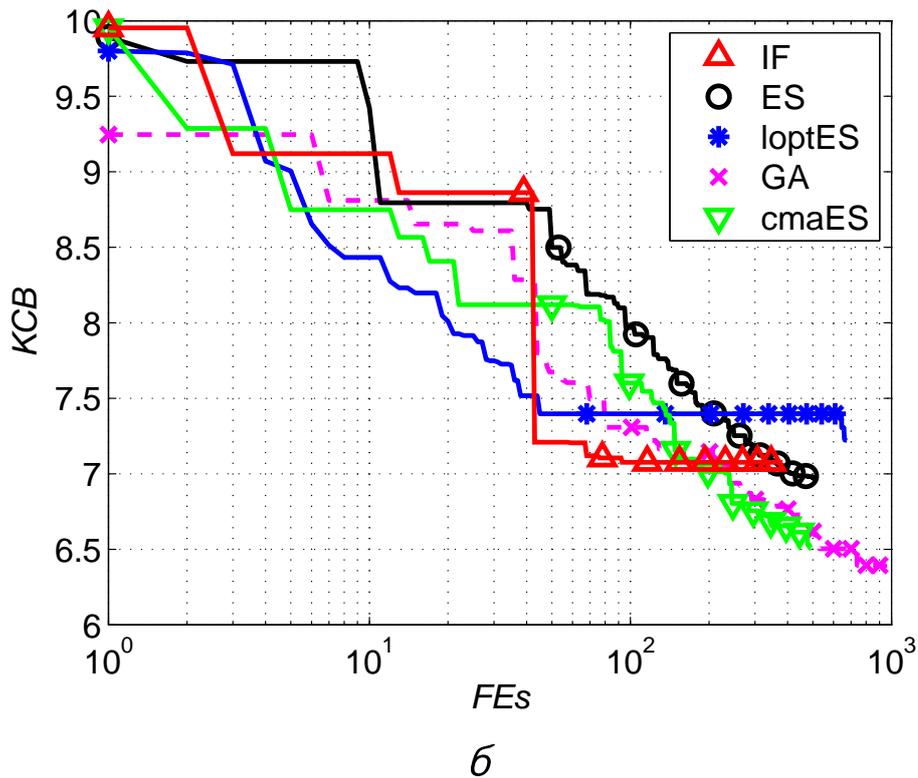
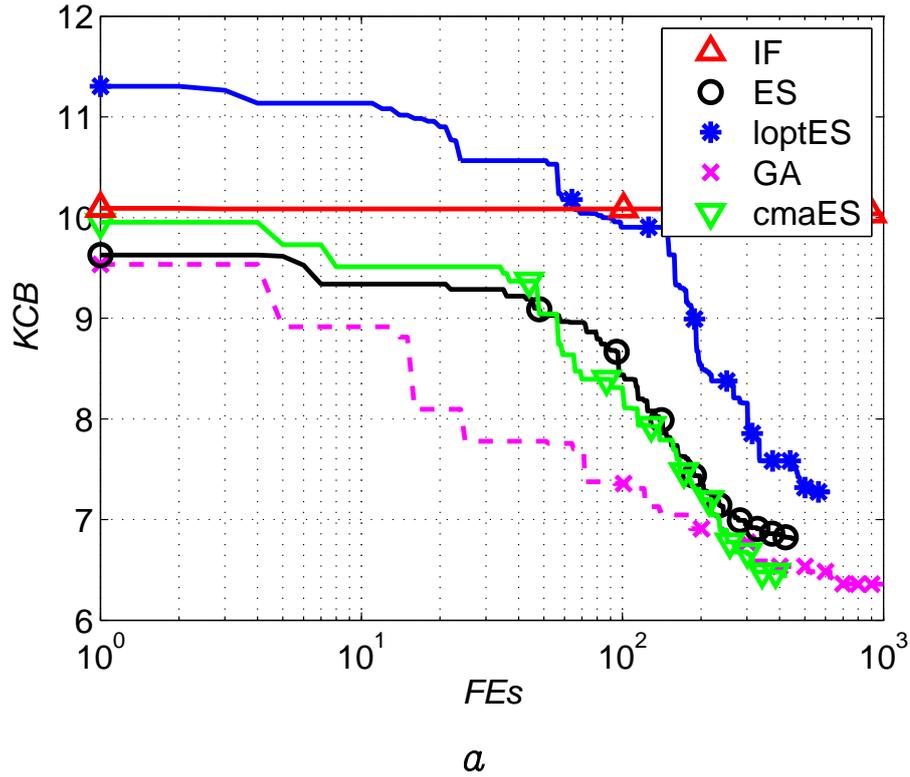


Рис. 4.9. Зависимость $\max(KCB)$ от количества вычислений целевой функции FEs с масштабированием параметров (а) и с функцией-декодером (б): \triangle – НФ; ∇ – $(\mu/\mu_W, \lambda)$ -АКМ-ЭС; \circ – $(\mu/\mu_I, \lambda)$ -сСА-ЭС; \times – ГА; $*$ – $(\lambda)_{opt}$ -сСА-ЭС

Для полного бюджета $FEs = 1000$ вычислений целевой функции лучшие результаты показывает ГА, который сходится к $\max(KCB)=6,39$. Алгоритм $(\mu/\mu_W, \lambda)$ -АКМ-ЭС достигает $\max(KCB)=6,5$ за $FEs = 492$.

При оптимизации широкодиапазонной быстроразворачиваемой модифицированной вибраторной антенны наименьшее значение $\max(KCB)=6,39$ получено с помощью ГА. Однако для его достижения ГА использует бюджет из 1000 вычислений целевой функции полностью, тогда как ЭС с адаптацией ковариационной матрицы сходится к $\max(KCB)=6,5$ за менее чем 500 вычислений целевой функции. Таким образом, с практической точки зрения для оптимизации рассмотренной задачи целесообразно использовать $(\mu/\mu_W, \lambda)$ -АКМ-ЭС, которой для поиска близких к оптимальным параметров вибраторной антенны требуется в 2 раза меньше вычислений целевой функции, чем ГА.

4.4. Оптимизация параметров математической модели резистора

Методики оптимизации ЭМС БА КА использовались для создания математической модели низкочастотного резистора, которая учитывает его поведение на частотах до 20 ГГц.

В качестве исходных данных (рис. 4.10) принимались результаты измерений для резистора P1-12-0.062 51 \pm 5 % Ом в безвыводном исполнении (тип корпуса 0603), проведенные на двух векторных анализаторах цепей: Wiltron 37369A (в диапазоне от 40 МГц до 20 ГГц) и Rohde-Schwarz ZVA40 (в диапазоне от 10 МГц до 20 ГГц) [3].

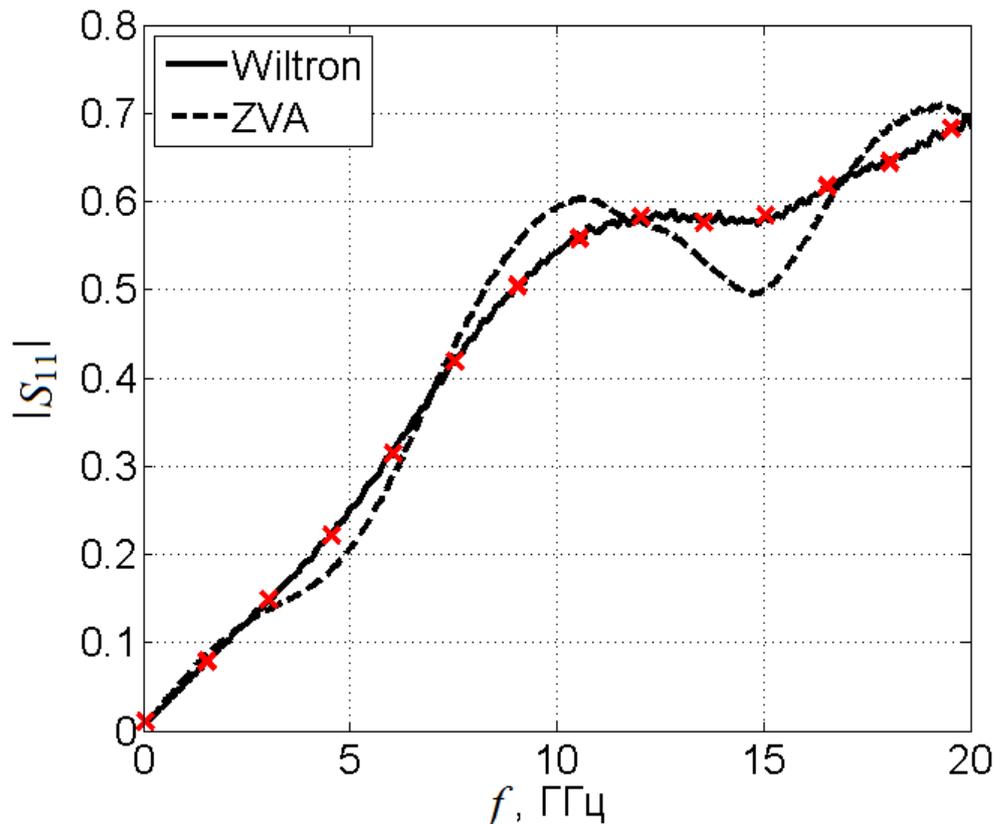


Рис. 4.10. Модуль коэффициента отражения S_{11} резистора P1-12-0.062 $51 \pm 5\%$ Ом в безвыводном исполнении (тип корпуса 0603): результаты измерений на Wiltron 37369A (сплошная линия) и Rohde-Schwarz ZVA40 (пунктирная линия)

Из двух экспериментальных кривых на рис. 4.10 для создания математической модели выбрана характеристика, полученная с помощью Wiltron 37369A (сплошная линия). Для ускорения оптимизации из 400 экспериментальных точек в сравнении с расчетной характеристикой эквивалентной схемы использовалась каждая 30-я точка (показаны на рис. 4.10 крестами). При необходимости количество учитываемых экспериментальных точек может быть увеличено путем изменения значения одной переменной в скрипте для моделирования.

Для получения эквивалентной модели резистора по аналогии с [114] выбрана схема, состоящая из 3 параллельно включенных RLC -звеньев (рис. 4.11).

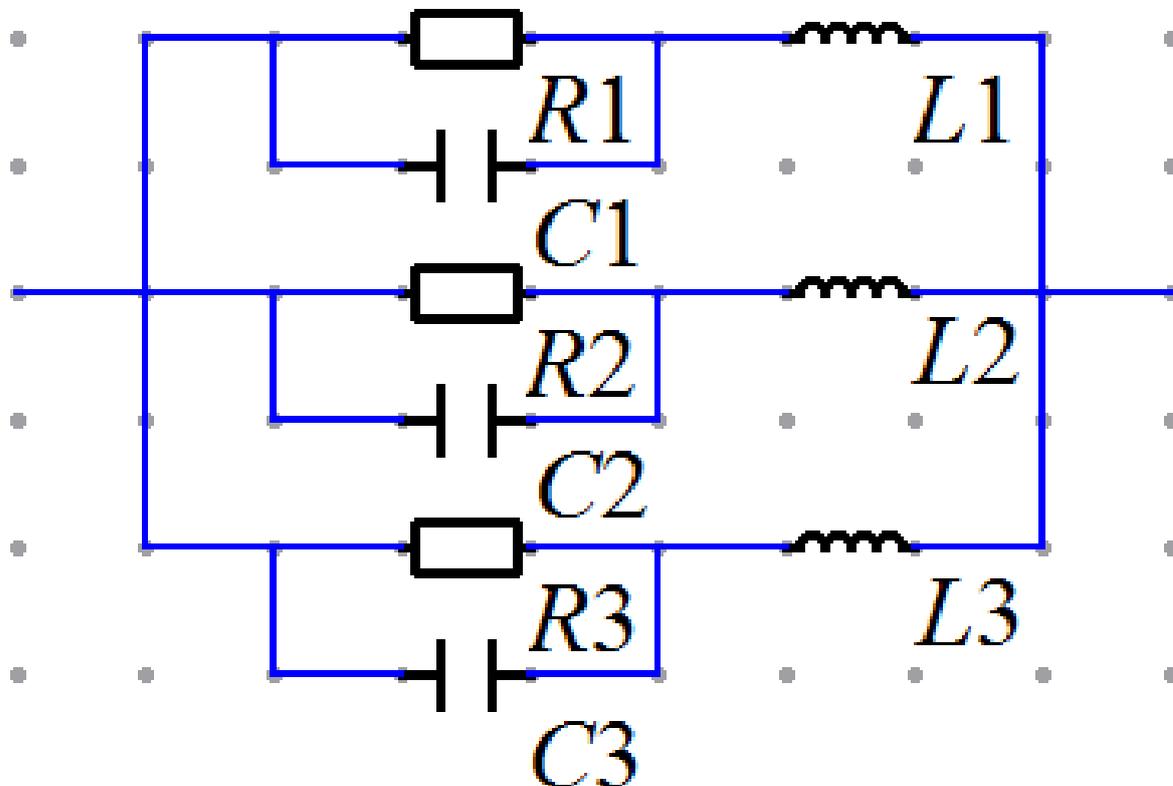
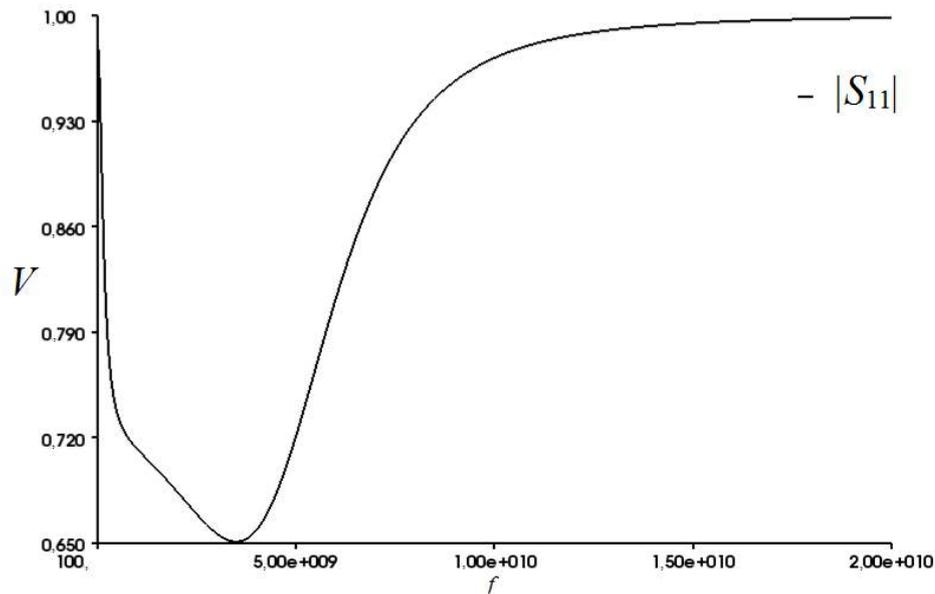


Рис. 4.11. Принципиальная схема эквивалентной модели резистора

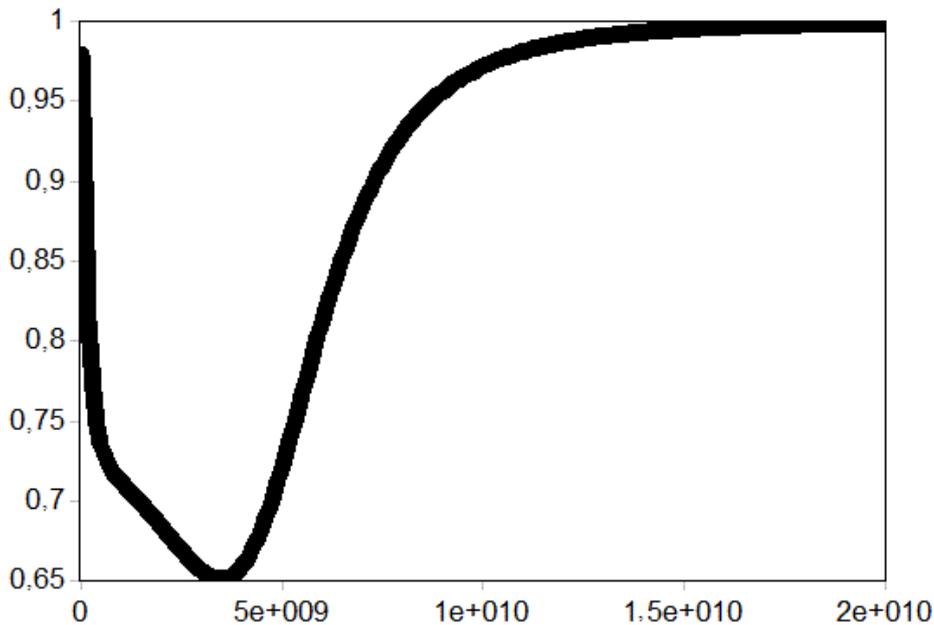
С целью проверки корректности моделирования системой TALGAT рассматриваемой эквивалентной модели резистора проведено сравнение результатов вычисления модуля коэффициента отражения S_{11} в системе TALGAT и ПО Sonnet [115] (рис. 4.12). При этом использовалась следующая выборка значений элементов эквивалентной модели: первое звено – $R_1 = 300$ Ом, $C_1 = 10^{-13}$ Ф, $L_1 = 9,5 \cdot 10^{-9}$ Гн; второе звено – $R_2 = 1$ Ом, $C_2 = 10^{-10}$ Ф, $L_2 = 10^{-7}$ Гн; третье звено – $R_3 = 1$ Ом, $C_3 = 6 \cdot 10^{-10}$ Ф, $L_3 = 9,5 \cdot 10^{-8}$ Гн.

Результаты расчета модуля коэффициента отражения $|S_{11}|$ для эквивалентной модели резистора, полученные

в системе TALGAT (рис. 4.12, *a*) и ПО Sonnet (рис. 4.12, *б*), совпадают, что позволяет сделать вывод о корректности результатов моделирования системой TALGAT.



a



б

Рис. 4.12. Результат расчета зависимости модуля коэффициента отражения $|S_{11}|$ для эквивалентной модели резистора от частоты (Гц) в системе TALGAT (*a*) и ПО Sonnet (*б*)

После проверки корректности моделирования на основе скрипта системы TALGAT для вычисления $|S_{11}|$ была реализована целевая функция, которая описывает задачу оптимизации

$$F(\mathbf{y}) = \sum_{i=1}^M (|S_{11}(\mathbf{y}_{\text{Wiltron}})|_i - |S_{11}(\mathbf{y})_i|)^2 \longrightarrow \min_{\mathbf{y}}, \quad (4.1)$$

где $|S_{11}(\mathbf{y}_{\text{Wiltron}})|_i$ – i -я экспериментальная точка $|S_{11}|$, полученная при измерениях с помощью Wiltron 37369A (показаны на рис. 4.10 крестами); $|S_{11}(\mathbf{y})_i|$ – соответствующая i -я точка $|S_{11}|$, вычисленная в системе TALGAT для эквивалентной модели резистора.

Таким образом, минимум целевой функции (4.1) равен нулю и соответствует эквивалентной модели резистора, вычисленный параметр $|S_{11}|$ которой совпадает с экспериментальными данными $|S_{11}|$ во всех точках.

Для оптимизации целевой функции (4.1) использовалась $(5/5_W, 10)$ -ЭС с адаптацией ковариационной матрицы и обработкой ограничений [59]. Примеры результатов оптимизации приведены на рис. 4.13, где знаки $+$ показывают $|S_{11}|$, вычисленный системой TALGAT для начальной точки поиска, \times – $|S_{11}|$ для экспериментальных данных (результаты измерений на Wiltron 37369A на рис. 4.10), \circ – $|S_{11}|$ для оптимальной эквивалентной модели, полученной после FES вычислений целевой функции.

На рис. 4.13, a параметр $|S_{11}(\mathbf{y}_{\text{opt}})|$ эквивалентной модели резистора (окружности) получен в результате оптимизации $(5/5_W, 10)$ -АКМ-ЭС с бюджетом 1500 вычислений целевой функции. $|S_{11}(\mathbf{y}_{\text{opt}})|$ совпадает с результатами измерений (кресты) для ряда частотных точек, соответствующее значение $F(\mathbf{y}_{\text{opt}}) = 7,76 \cdot 10^{-3}$, значения параметров:

$R_1 = 78,2 \text{ Ом}, C_1 = 2,19 \cdot 10^{-13} \text{ Ф}, L_1 = 2,26 \cdot 10^{-9} \text{ Гн},$
 $R_2 = 233,5 \text{ Ом}, C_2 = 10^{-14} \text{ Ф}, L_2 = 10^{-10} \text{ Гн}, R_3 = 235,12 \text{ Ом},$
 $C_3 = 3,43 \cdot 10^{-12} \text{ Ф}, L_3 = 6 \cdot 10^{-8} \text{ Гн}.$

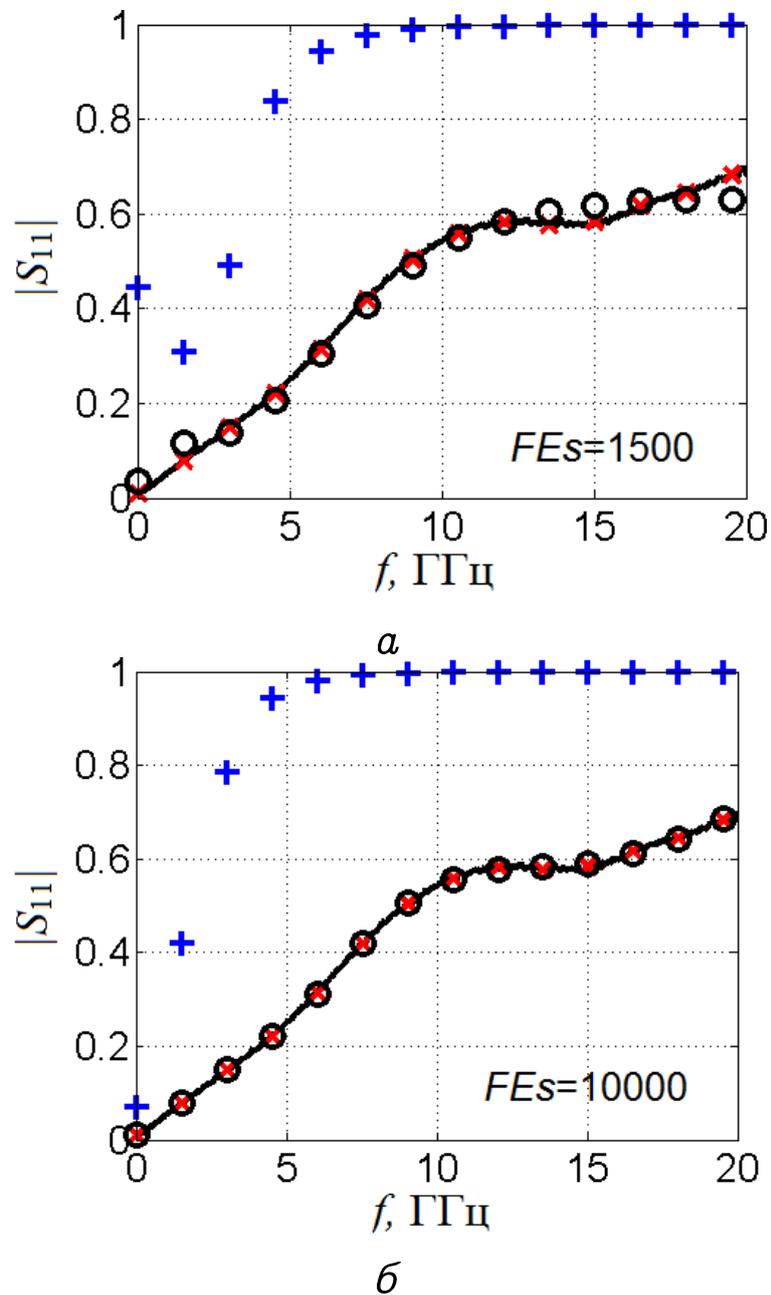


Рис. 4.13. Модуль коэффициента отражения $|S_{11}|$ для эквивалентной модели резистора, полученной в результате оптимизации $(5/5_W, 10)$ -АКМ-ЭС при 1500 (а) и 10^4 (б) вычислений целевой функции (4.1): \times – $|S_{11}|$ для экспериментальных данных; $+$ – $|S_{11}(y_{init})|$ для начальной точки поиска; \circ – $|S_{11}(y_{opt})|$ для оптимальной эквивалентной модели

Алгоритм $(5/5_W, 10)$ -ЭС использовал бюджет из 10000 вычислений целевой функции и получил $|S_{11}(y_{\text{opt}})|$, который совпадает с результатами измерений для всех частот до 20 ГГц ($F(y_{\text{opt}}) = 2,23 \cdot 10^{-4}$, значения параметров: $R_1 = 139,8$ Ом, $C_1 = 1,53 \cdot 10^{-13}$ Ф, $L_1 = 3,14 \cdot 10^{-9}$ Гн, $R_2 = 306,76$ Ом, $C_2 = 3,17 \cdot 10^{-14}$ Ф, $L_2 = 2 \cdot 10^{-9}$ Гн, $R_3 = 108,98$ Ом, $C_3 = 1,54 \cdot 10^{-13}$ Ф, $L_3 = 5,53 \cdot 10^{-5}$ Гн). Следовательно, $(\mu/\mu_W, \lambda)$ -АКМ-ЭС при увеличении количества вычислений целевой функции сходится к оптимуму целевой функции (4.1) и позволяет получить параметры эквивалентной модели резистора по заданной частотной зависимости модуля коэффициента отражения $|S_{11}|$.

Рассмотренная здесь целевая функция (4.1) имеет фиксированное количество параметров – 9. Далее показана возможность оптимизации ЭС целевой функции с количеством параметров от 7 до 19.

4.5. Оптимизация параметров многопроводной линии передачи

Выполним оптимизацию параметров поперечного сечения многопроводной линии передачи (МПЛП) в неоднородном диэлектрическом заполнении, которая используется в печатных платах УЭМ ВА КА. МПЛП имеет фиксированную ширину w_{total} и состоит из M проводников, расположенных на слое диэлектрика на заземленном слое. Каждый i -й проводник имеет ширину $w_i \in \mathbb{R}$ и отделен от $(i + 1)$ -го проводника расстоянием $s_i \in \mathbb{R}$, $i = 0, \dots, M$. С помощью системы TALGAT для заданной пары векторов $(\mathbf{w}, \mathbf{s}) = (w_1, w_2, \dots, w_M, s_0, s_1, s_2, \dots, s_M)^T$, которые

содержат ширину проводников и расстояния между ними, можно рассчитать матрицу характеристического импеданса $\mathbf{Z}_0(\mathbf{w}, \mathbf{s})$ размерностью $M \times M$. Цель оптимизации – выбрать (\mathbf{w}, \mathbf{s}) таким образом, чтобы главная диагональ матрицы $\mathbf{Z}_0(\mathbf{w}, \mathbf{s})$ содержала значения импеданса, как можно более близкие к оптимальному импедансу Z_{opt} . Для этого необходимо минимизировать сумму разностей $\sum_{i=1}^M |Z_{ii}(\mathbf{w}, \mathbf{s}) - Z_{\text{opt}}|$, где $Z_{ii}(\mathbf{w}, \mathbf{s})$ – диагональные элементы матрицы $\mathbf{Z}_0(\mathbf{w}, \mathbf{s})$. Каждый проводник должен иметь неотрицательную ширину $w_i \geq \check{w} > 0$ и расстояние $s_i \geq \check{s} > 0$. Ширина всех проводников вместе с расстояниями между ними должны давать в сумме общую ширину МПЛП w_{total} . Кроме того, необходимо уменьшить емкостную и индуктивную связь между проводниками, что соответствует минимизации внедиагональных элементов матрицы \mathbf{Z}_0 . Таким образом, получаем следующую задачу оптимизации для вектора параметров $(\mathbf{w}, \mathbf{s}) \in \mathbb{R}^N$:

$$F_1(\mathbf{w}, \mathbf{s}) = \sum_{i=1}^M |Z_{ii}(\mathbf{w}, \mathbf{s}) - Z_{\text{opt}}| \longrightarrow \min_{\mathbf{w}, \mathbf{s}}, \quad (4.2)$$

так что

$$\check{w} \leq w_j, \check{s} \leq s_j, \quad (4.3)$$

$$\sum_{j=1}^M w_j + \sum_{j=0}^M s_j = w_{\text{total}}, \quad (4.4)$$

$$F_2(\mathbf{w}, \mathbf{s}) = \sum_{i=1}^M \sum_{j=1, j \neq i}^M Z_{ij}(\mathbf{w}, \mathbf{s}) \longrightarrow \min_{\mathbf{w}, \mathbf{s}}. \quad (4.5)$$

Размерность пространства поиска для задачи (4.2)–(4.5) $N = 2M + 1$.

Выражения (4.2)–(4.5) описывают задачу оптимизации с двумя критериями, которая может быть решена либо с по-

мощью средств многокритериальной оптимизации (таких как популяционные не основанные на Парето, основанные на Парето и алгоритмы определения ниш [116]), либо путем скаляризации задачи. Мы применим второй способ и объединим критерии (4.2) и (4.5) в одну агрегированную целевую функцию

$$F_{\alpha}(\mathbf{w}, s) = \alpha F_1(\mathbf{w}, s) + (1 - \alpha) F_2(\mathbf{w}, s), \quad (4.6)$$

где $0 \leq \alpha \leq 1$ – весовой коэффициент, значение которого определяется относительной важностью рассматриваемых критериев. Агрегированные целевые функции, подобные (4.6), являются вычислительно эффективными [117], но их значения зависят от выбора α . Вопрос выбора правильного значения коэффициента α рассматривается ниже.

Так как количество проводников МПЛП не ограничено, размерность пространства поиска N задачи (4.6) может быть бесконечно большой. Для случаев нескольких проводников этот факт не играет роли, однако для более сложных МПЛП необходим алгоритм оптимизации, который эффективно работает с большими N . Выбранный алгоритм должен также правильно обрабатывать граничные условия в целевой функции. Этим требованиям удовлетворяет $(\mu/\mu_I, \lambda)$ -АКМ- σ СА-ЭС с ограничениями.

Вариант $(\mu/\mu_I, \lambda)$ -АКМ- σ СА-ЭС с ограничениями был предложен в [65] для решения задачи оптимизации в области финансов. Так как формулировка этой успешно решенной задачи сходна с задачей (4.2)–(4.5), для решения последней может быть также использована σ -самоадаптационная АКМ-ЭС с ограничениями. Эволюционные алгоритмы уже применялись для решения задач оптимизации МПЛП [118], но только для случая простой конфигурации с одним

проводником. Разные эволюционные алгоритмы, в том числе ЭС, применялись также для решения задач ЭМС и проектирования антенн [119].

Рассмотрим оптимизацию МПЛП с помощью $(\mu/\mu_I, \lambda)$ -АКМ-сСА-ЭС поэтапно.

Математическая модель. Рассматриваемая конфигурация МПЛП встречается на печатных платах УЭМ БА КА. В качестве примера на рис. 4.14 показана МПЛП с $M = 2$ проводниками на диэлектрическом слое. Покрывающая МПЛП паяльная маска не показана.

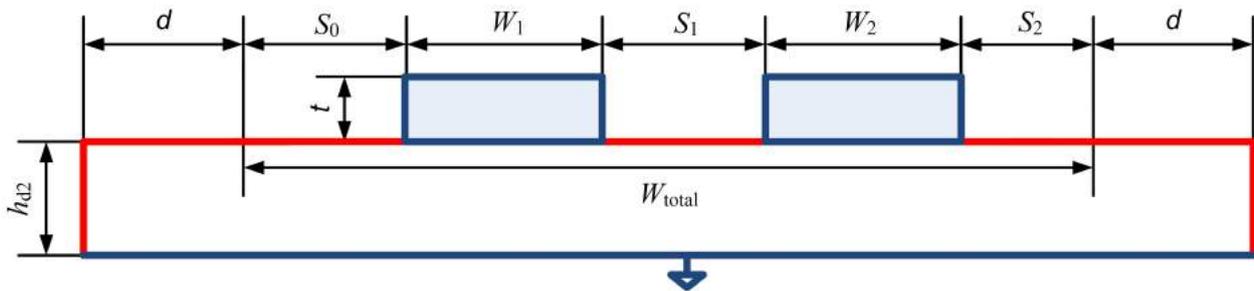


Рис. 4.14. МПЛП с $M = 2$ проводниками. Паяльная маска не показана

В конфигурации на рис. 4.14 необходимо оптимизировать ширину проводников w_1 и w_2 и соответствующие расстояния s_0 , s_1 и s_2 между проводниками, принимая во внимание равенство (4.4). Все другие параметры зафиксированы: толщина проводника $t = 0,065$ мм, толщина паяльной маски $h_{d1} = 0,02$ мм (не показана на рис. 4.14), толщина препрега $h_{d2} = 0,1446$ мм, расстояние от края МПЛП $d = 1$ мм, относительная диэлектрическая проницаемость паяльной маски $\epsilon_{r1} = 3,5$ и препрега $\epsilon_{r2} = 4,49$, тангенс угла диэлектрических потерь паяльной маски $\tan \delta_1 = 0,03$ и препрега $\tan \delta_2 = 0,014$.

Для заданного вектора параметров $(\mathbf{w}, \mathbf{s}) = (w_1, w_2, s_0, s_1, s_2)^T$ вычисляется матрица характеристического импеданса $\mathbf{Z}_0(\mathbf{w}, \mathbf{s})$ с помощью метода моментов.

Целевая функция. При расчете целевой функции необходимо принять во внимание несколько факторов.

Во-первых, на концах МПЛП находятся цепи с заданным импедансом, который определяет оптимальное значение Z_{opt} . В экспериментах задано типовое значение $Z_{\text{opt}} = 50$ Ом, однако предложенный алгоритм может использоваться для любого значения Z_{opt} .

Во-вторых, минимально возможные значения \check{w} и \check{s} в условии (4.3) зависят от технологического процесса, по которому производится печатная плата. Для рассматриваемой печатной платы принято $\check{w} = \check{s} = 0,1$ мм. Общая ширина МПЛП $w_{\text{total}} = 5$ мм. Так как ЭС в алгоритме 1.4 работает с векторами параметров \mathbf{y} с компонентами $0 \leq y_i \leq 1$ и $\sum_{i=1}^N y_i = 1$, необходимо использовать функцию-декодер для преобразования \mathbf{y} в (\mathbf{w}, \mathbf{s}) в соответствии с правилами $w_i = \check{w} + \beta y_i$ и $s_i = \check{w} + \beta y_{M+i+1}$, где коэффициент преобразования $\beta = w_{\text{total}} - \check{w}N$, $N \neq w_{\text{total}}/\check{w}$ выводится путем вставки правил в формулу (4.4), при этом $\check{w} = \check{s}$ и $N = 2M + 1$. Таким образом, функция-декодер записывается как

$$(\mathbf{w}, \mathbf{s}) = \text{decode}(\mathbf{y}) := (\check{w} + \beta y_1, \dots, \check{w} + \beta y_N)^T. \quad (4.7)$$

В-третьих, агрегированная целевая функция $F_\alpha(\mathbf{w}, \mathbf{s})$ (4.6) зависит от весового коэффициента α , который определяет приоритет критериев оптимизации F_1 и F_2 . Правильно выбранное значение α должно обеспечивать одновременно

сходимость по критерию F_1 (то есть разница (4.2) должна уменьшаться в каждом поколении) и минимально возможное значение критерия F_2 .

Чтобы выбрать корректное значение α , исследовались зависимости F_1 и F_2 от α . Алгоритм σ -самоадапционной АКМ-ЭС с ограничениями и $\mu = 3$, $\lambda = 12$ запускался для $\alpha = 0; 0,2; \dots; 1$, начальной интенсивности мутации $\sigma^{(0)} = 1/\sqrt{N}$ и случайно выбранного $y^{(0)}$. Записывались минимальные значения F_1 и F_2 , при которых алгоритм завершался (условия завершения: $\sigma^{(g)} < 10^{-6}$ или $g = 10^3$). Усредненные по 10 независимым запускам результаты для $M = 3$ и $M = 5$ приведены на рис. 4.15, где точками показаны средние значения F_1 и F_2 .

При оптимизации МПЛП с 3 проводниками (рис. 4.15, а) для $\alpha = 0$ критерий F_1 полностью игнорируется и сохраняет большие значения, тогда как ЭС фокусируется на втором критерии и находит минимальное значение F_2 . С ростом α ЭС достигает меньших значений F_1 , однако изменения в F_2 менее выражены. Значительный рост F_2 наблюдается для экстремального случая $\alpha = 1$, когда ЭС игнорирует второй критерий. Случай $M = 5$ (рис. 4.15, б) качественно не отличается от случая $M = 3$. Принимая во внимание требование получить минимально возможное значение F_2 , выберем $\alpha = 0,2$, при котором ЭС достигает значения F_2 , близкого к минимальному $F_2(0)$, и в то же время обеспечивает сходимость по критерию F_1 . Проиллюстрируем результаты оптимизации параметров МПЛП с числом проводников $M = 3$ и выбранным значением $\alpha = 0,2$.

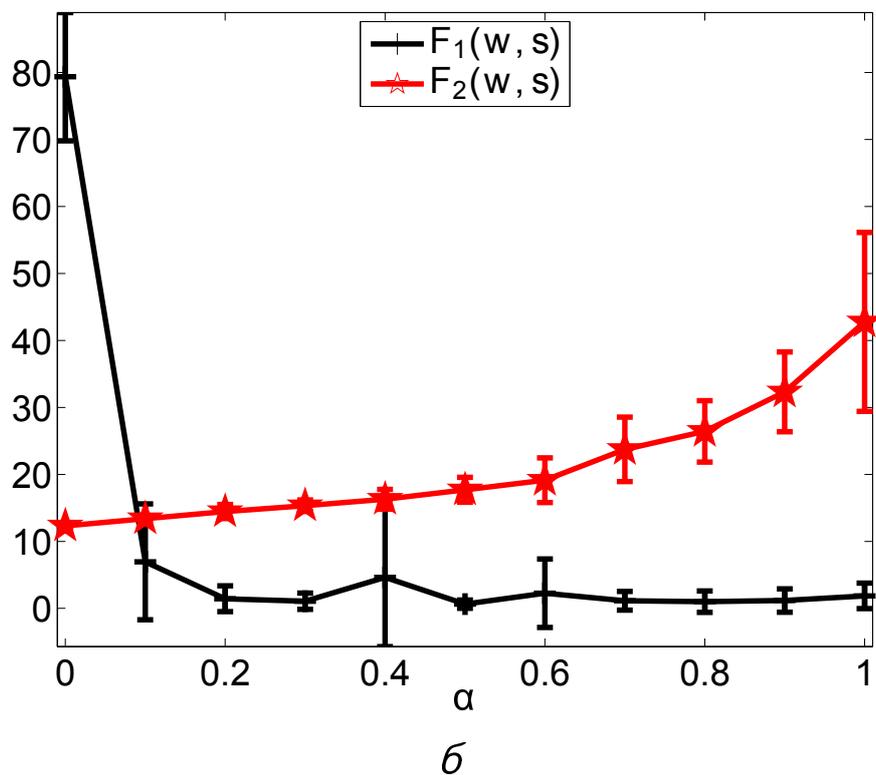
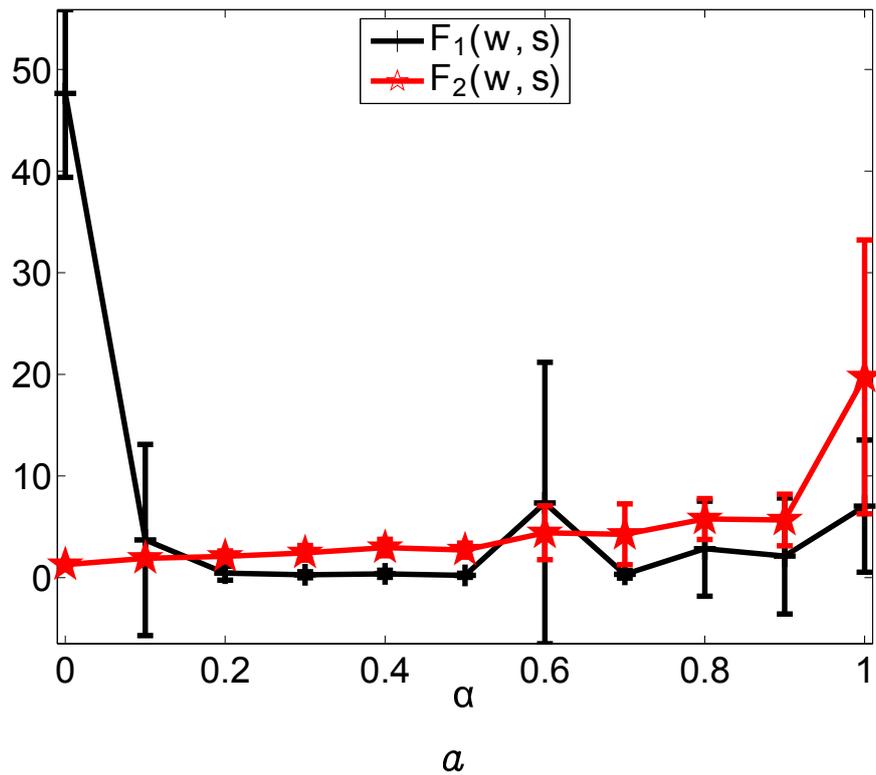


Рис. 4.15. Значения $F_1(w, s)$ и $F_2(w, s)$, достигнутые (3/3, 10)-АКМ- σ СА-ЭС после 100 поколений: а - $M = 3$; б - $M = 5$

Результат оптимизации после 100 поколений с приведенными выше параметрами ЭС показан на рис. 4.16.

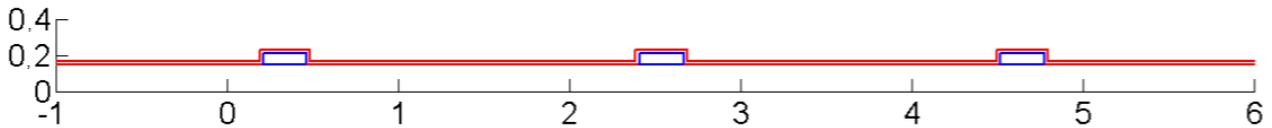


Рис. 4.16. Оптимальная МПЛП с $M = 3$. Шкала в миллиметрах

Оптимальная МПЛП имеет $F_\alpha(\mathbf{w}, \mathbf{s}) = 1,32$, $F_1(\mathbf{w}, \mathbf{s}) = 0,07$ и $F_2(\mathbf{w}, \mathbf{s}) = 1,63$. Фиксированное расстояние $d = 1$ мм расположено с обеих сторон МПЛП при $x < 0$ и $x > 5$ мм. Все проводники имеют одинаковую ширину $0,26$ мм, расстояние между ними $s_1 = 1,94$ мм и $s_2 = 1,85$ мм. Расстояния по краям МПЛП $s_0 = 0,21$ мм и $s_3 = 0,22$ мм дополняют сумму всех размеров до общей ширины $w_{\text{total}} = 5$ мм.

Расположение проводников, определяемое парой векторов (\mathbf{w}, \mathbf{s}) , может меняться при повторных запусках алгоритма, так как оптимальное значение Z_{opt} может быть достигнуто при выборе разных конфигураций МПЛП, а целевая функция $F_\alpha(\mathbf{w}, \mathbf{s})$ не содержит ограничений в отношении расположения проводников. В целевой функции также нет прямых ограничений на симметрию МПЛП, но синтезируемые конфигурации МПЛП имеют тенденцию к симметричности, так как у них меньшее значение F_2 . Таким образом, ЭС воспроизводит конфигурации МПЛП, которые выбирают эксперты [124]: они обычно рассчитывают ширину проводников с помощью приближенной аналитической формулы для линий передачи с одним проводником и затем распределяют проводники равномерно, получая симметричные МПЛП.

Исследование производительности эволюционной стратегии. Реальная печатная плата, на основе которой сформулирована задача (4.2)–(4.5), содержит несколько МПЛП с $M = 2, 3, 4, 8, 9$. Пропустим тривиальный случай $N = 5$ (как было уже сказано, $N = 2M + 1$) и проведем по 100 экспериментов с (3/3, 12)-АКМ-σСА-ЭС для $N = 7, 9, \dots, 19$ при тех же условиях, что описаны в предыдущем пункте. Так как F_α и F_2 сходятся менее чем за 100 поколений и для них невозможно собрать данные по масштабированию производительности, исследуем критерий F_1 . Для обработки экспериментальных данных необходимо определить эталонное значение F_{ref} , которое аппроксимирует глобальный минимум F_1 . С этой целью выберем потомка с наименьшим значением F_1 из всех потомков, созданных в течение 100 экспериментов, и используем это значение F_1 как F_{ref} . Выбранные значения F_{ref} для разных N приведены в табл. 4.2.

Таблица 4.2

F_{ref} для $\alpha = 0, 2$

N	7	9	11	13	15	17	19
F_{ref} , кОм	0,050	0,066	0,348	0,571	0,609	1,147	17,852

Заметим, что ЭС не достигает F_{ref} при каждом запуске, так как задача (4.2)–(4.5) является многоэкстремальной, что можно обнаружить путем построения графика зависимости разности $\Delta F_1 = (F_1)_{\text{best}} - F_{\text{ref}}$ от $\Delta y = \|y_{\text{best}} - y_{\text{ref}}\|$ (рис. 4.17, а), где $(F_1)_{\text{best}}$ – наименьшее значение F_1 , достигнутое в одном запуске ЭС, y_{best} и y_{ref} – векторы параметров, относящиеся соответственно к $(F_1)_{\text{best}}$ и F_{ref} . Наименьшее значение F_1 отдельных запусков ЭС отличается от F_{ref} максимум на 2,5 Ом для $N = 9$ и на 7 Ом для $N = 19$, тогда

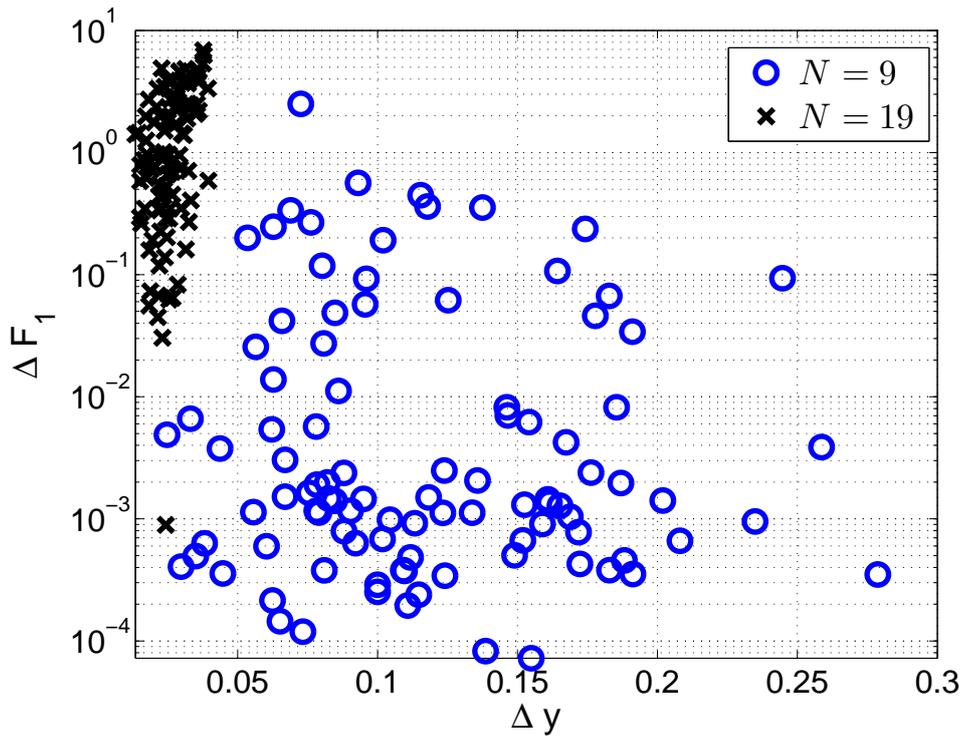
как расстояние Δy также значительно варьируется, что указывает на существование большого количества локальных оптимумов в различных точках пространства поиска.

Для построения кривых масштабирования производительности ЭС рассмотрим последовательность $F_{\text{target}}(k) = F_{\text{ref}} + 10^k$, $k = 2, 1, \dots, -4$, и измерим количество вычислений целевой функции $\#FE(k)$, которое ЭС затрачивает для достижения $F_1^{(g)} \leq F_{\text{target}}(k)$. Каждый раз, когда ЭС достигает $F_{\text{target}}(k)$, инкрементируется счетчик успешных запусков $\#\text{succ}(k)$. Суммирование $\#FE(k)$ по всем 100 запускам и вычисление ожидаемого времени оптимизации (expected running time, ERT) по формуле [120]

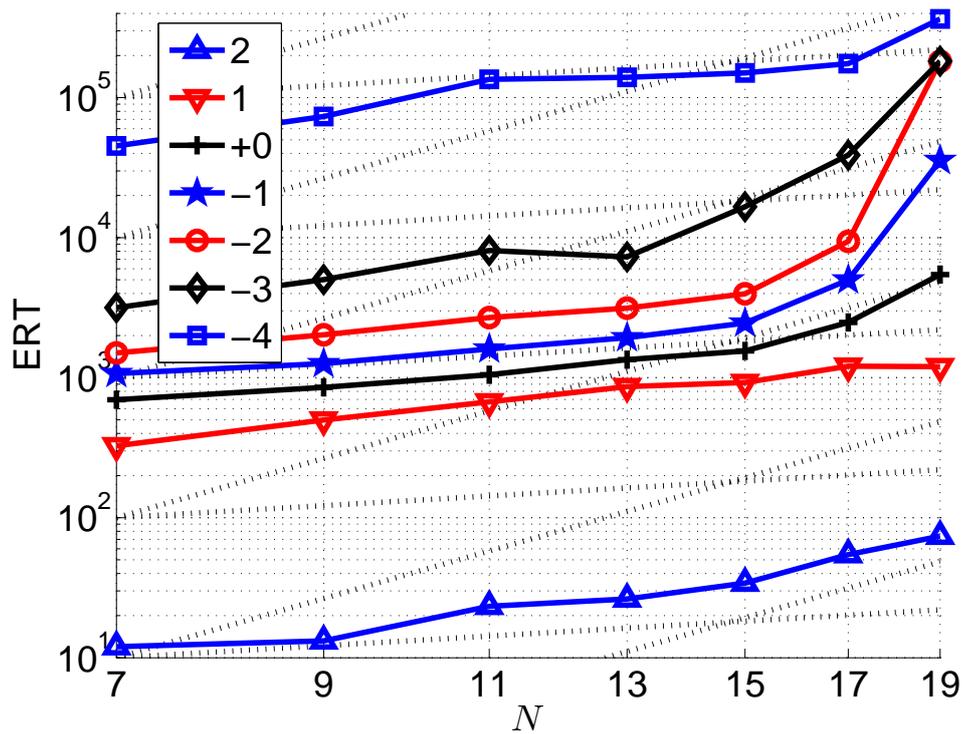
$$\text{ERT}(k) = \sum_{r=1}^{100} \#FE_r(k) / \#\text{succ}(k) \quad (4.8)$$

дает в итоге оценку ожидаемого времени оптимизации. Полученные кривые представлены на рис. 4.17, б.

Они показывают оценку ожидаемого числа вычислений целевой функции, которое необходимо затратить, чтобы достичь глобального минимума F_{ref} с заданной точностью. Нижняя кривая соответствует точности $F_{\text{ref}} + 10^2$, которой ЭС достигает меньше чем за 70 вычислений целевой функции, так как начальный случайно выбранный вектор параметров $y^{(0)}$ ведет к изначально близким к 10^2 значениям F_1 . Вторая кривая снизу представляет уровень точности $F_{\text{ref}} + 10$, который требует не менее 280 вычислений целевой функции, так как ЭС тратит их на адаптацию интенсивности мутации. После адаптации оптимальной интенсивности мутации σ следуют уровни точности $F_{\text{ref}} + 10^{\{0, -1, -2\}}$. Кривая $F_{\text{ref}} + 1$ (кресты) масштабируется для $N = 7, \dots, 15$ линейно с N .



a



б

Рис. 4.17. Эксперименты с (3/3, 12)-АКМ-σСА-ЭС для $\alpha = 0, 2$: зависимость $\Delta F_1 = (F_1)_{\text{best}} - F_{\text{ref}}$ от $\Delta y = \|y_{\text{best}} - y_{\text{ref}}\|$ для $N = 9, 19$ (*a*) и ожидаемое время оптимизации для достижения $F_1^{(g)} \leq F_{\text{ref}} + 10^{\{2, 1, 0, -1, \dots, -4\}}$ как функция N (*б*)

Таким образом, ЭС сходится к окрестности F_{ref} примерно за $N \cdot 10^2$ вычислений целевой функции. Кривая $F_{\text{ref}} + 1$ растет квадратично для $N = 19$, так как количество успешных запусков уменьшается с 91 для $N = 17$ до 38 для $N = 19$. То же самое наблюдается для кривых $F_{\text{ref}} + 10^{-1, -2}$.

Уровень точности $F_{\text{ref}} + 10^{-3}$ (ромбы) можно рассматривать как переходную фазу к кривой $F_{\text{ref}} + 10^{-4}$, которая превышает заданный бюджет из 10^3 поколений ($1, 2 \cdot 10^4$ вычислений целевой функции) для всех $N > 7$, то есть ЭС не достигает $F_1 \leq F_{\text{ref}} + 10^{-4}$ в большинстве экспериментальных запусков.

В целом ЭС достигает точности $F_1 \leq F_{\text{ref}} + 0,1$ для $N = 7, \dots, 19$ меньше чем за $1, 2 \cdot 10^4$ вычислений целевой функции. Этот уровень точности соответствует оптимальным конфигурациям МПЛП с $|Z_{ii} - Z_{\text{opt}}| \ll 1$ Ом и является удовлетворительным результатом с практической точки зрения. Таким образом, ЭС корректно решает задачу (4.2)–(4.5) и выбирает оптимальные параметры (w, s) , которые минимизируют $F_1(w, s)$ и, следовательно, агрегированную целевую функцию $F_\alpha(w, s)$.

4.6. Структурная и параметрическая оптимизация многокаскадного модального фильтра

В процессе проектирования модальных фильтров особую важность имеет задача поиска оптимальной конструкции фильтра, которая позволяла бы получить требуемые характеристики и удовлетворяла дополнительным требованиям, таким как простота и надежность. Рассмотрим модальный фильтр [121–123] с фиксированными нагрузками в активном проводнике $R_{\text{акт}} = 98,4584$ Ом (рис. 4.18).

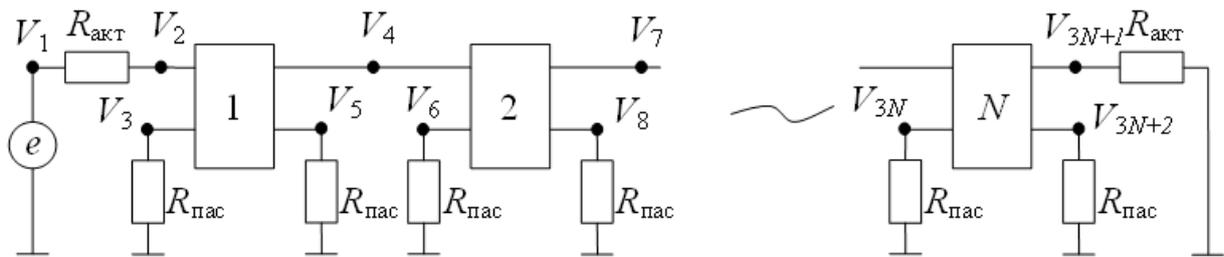


Рис. 4.18. Семикаскадный модальный фильтр ($N = 7$)

Задача оптимизации состоит в нахождении такого набора нагрузок $R_{\text{пас}}$ линий передачи, при котором полученный модальный фильтр удовлетворяет для узла 22 (см. рис. 4.18) двум критериям:

- 1) амплитуда напряжения минимальна;
- 2) АЧХ не имеет резонансов до 100 МГц.

Параметрами оптимизации являются сопротивления 14 пассивных резисторов $R_{\text{пас}}$. С целью упрощения и снижения стоимости конструкции выбран вариант, при котором имеются только два значения $R_{\text{пас}}$: короткое замыкание и холостой ход. В целевой функции они обозначены соответственно нулем и единицей, что позволяет кодировать сопротивление всех 14 резисторов в виде одного 14-битного двоичного числа. Затем данное число можно либо напрямую использовать в генетическом алгоритме, либо преобразовать в действительное представление для использования в алгоритмах, работающих с пространствами поиска из действительных чисел.

Если оптимизация ведется с действительными числами, то внутри целевой функции выполняется декодирование 14 битов числа в набор значений для резисторов: если первый бит 0, то значение сопротивления первого резистора соответствует короткому замыканию, иначе – холостому ходу и т.д. Для полученного модального фильтра вычисляется

временной отклик и амплитудно-частотная характеристика для диапазона $0, \dots, 105$ МГц с шагом 1 МГц посредством модуля вычисления отклика RESPONSE. По временному отклику выполняется проверка первого критерия оптимизации – минимальности амплитуды напряжения в узле 22, а по АЧХ – проверка второго критерия. Амплитуда напряжения при этом становится значением целевой функции, которое непосредственно минимизирует алгоритм оптимизации. Вторым критерий вводится в значение целевой функции в виде штрафного условия – если до 100 МГц имеется хотя бы один резонанс, то значение целевой функции умножается на 1000. Это делает его заведомо большим, чем значения целевой функции для модальных фильтров без резонансов до 100 МГц (значения целевой функции для них имеют порядок не более десятков вольт), и алгоритм оптимизации отбрасывает такие решения.

Для тестирования правильности представленной процедуры оптимизации выполнена ее реализация в системе TALGAT. Вся процедура описывается в виде одного файла на языке скриптов TALGAT и состоит из следующих этапов.

1. Расчет матриц параметров R, L, C, G для 7 однотипных линий передачи в модальном фильтре с помощью модуля анализа двумерных конфигураций MOM2D.
2. Задание целевой функции в виде динамической команды.
3. Вызов алгоритма оптимизации, которому в качестве параметра передается имя динамической команды с целевой функцией.

В качестве простейшего алгоритма оптимизации использован полный перебор, так как для 14 резисторов с 2 возможными значениями сопротивления число комбинаций составляет всего лишь 16383. На компьютере класса Pentium 4 (3 ГГц) полный перебор 16383 комбинаций занял 50 ч. В результате найдена следующая оптимальная комбинация значений сопротивлений: хх (холостой ход) – кз (короткое замыкание) – хх – кз – хх – кз – кз – хх – кз – хх – кз – хх – хх – хх. При этом на основе экспертной оценки конструкции модального фильтра была предложена комбинация кз – хх – кз – хх . . . – хх, для которой максимальная амплитуда напряжения в узле 22 $\max(V_{22}) = 6,74$ В. Для оптимальной комбинации, полученной перебором, $\max(V_{22}) = 6,02$ В. Так как для ускорения вычисления целевой функции характеристики модального фильтра вычислялись с пониженной точностью, данные значения находятся в пределах погрешности расчетов друг от друга. Таким образом, с помощью описанной целевой функции получен модальный фильтр с теми же характеристиками, что и у фильтра, подобранного экспертом. Это говорит о корректности реализации целевой функции.

АЧХ модального фильтра, полученного посредством оптимизации перебором, представлена на рис. 4.19. Видно, что оптимальный модальный фильтр не удовлетворяет второму критерию для оптимизации – АЧХ имеет 2 резонанса в полосе до 100 МГц. Действительно, максимальная амплитуда напряжения в узле 22 $\max(V_{22})$ составляет 6,02 В, в то время как значение целевой функции для этого фильтра равно 6020,22. Это говорит о том, что штрафное условие из-за наличия резонансов до 100 МГц выполнилось и целевая функция была умножена на 1000. Однако данный

модальный фильтр все равно является лучшим решением поставленной задачи, так как анализ файла с данными о результатах полного перебора показывает, что второму критерию не удовлетворяет ни одна из 16383 комбинаций.

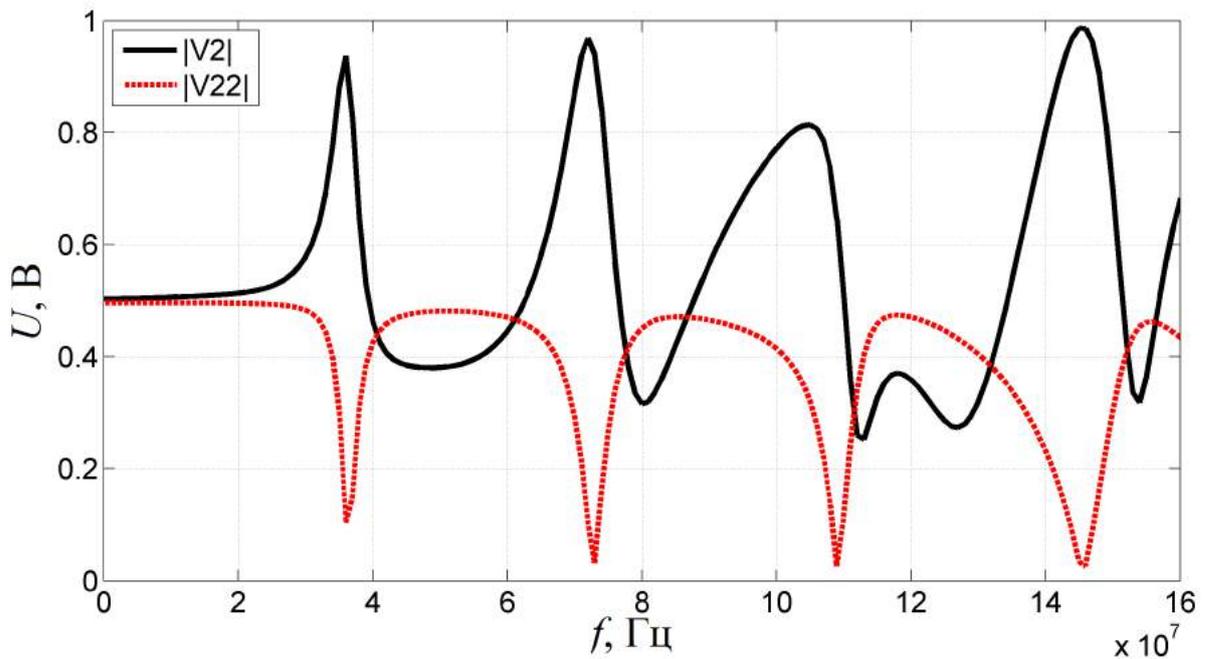


Рис. 4.19. АЧХ оптимального модального фильтра, полученного методом перебора

Рассмотрим случай параметрической оптимизации: применим к поставленной задаче ЭС с ограничениями (см. алгоритм 1.4). Для этого необходимо по аналогии с выражением (4.7) записать функцию-декодер. Минимально возможное значение $R_{\min} = 0,1$ Ом (короткое замыкание), максимально возможное — $R_{\max} = 10^6$ Ом (холостой ход). Верхняя граница R_{\max} кодируется следующим образом: задается фиксированная сумма сопротивлений $R_{\text{total}} = R_{\max} (M + 1)$, где $M = 14$ — количество резисторов. Затем вектор y преобразуется в вектор сопротивлений $(R_1, R_2, \dots, R_M, R_{M+1})$ по правилу $R_i = R_{\min} + \beta y_i$, где коэффициент преобразования $\beta = R_{\text{total}} - R_{\min} (M + 1)$, а значение R_{M+1} исполь-

зуются как «виртуальный» резистор, который обеспечивает выполнение условия $\sum_{i=1}^N y_i = 1$ и позволяет ЭС с ограничениями из алгоритма 1.4 выбирать для реальных резисторов $R_i, i = 1, \dots, M$, произвольные значения в диапазоне $R_{\min}, \dots, R_{\max}$.

Запуск 1000 поколений (3/3_I, 12)-АКМ-σСА-ЭС с ограничениями для описанной выше целевой функции завершился на поколении 644 со следующим результатом: $(R_1, R_2, \dots, R_M, R_{M+1}) = (11552,05; 0,10; 7368,59; 0,10; 0,10; 3038,05; 27914,69; 0,10; 0,10; 9949,70; 736,44; 0,10; 76,26; 2333,05; 87030,55)$ Ом. АЧХ полученного модального фильтра показана на рис. 4.20.

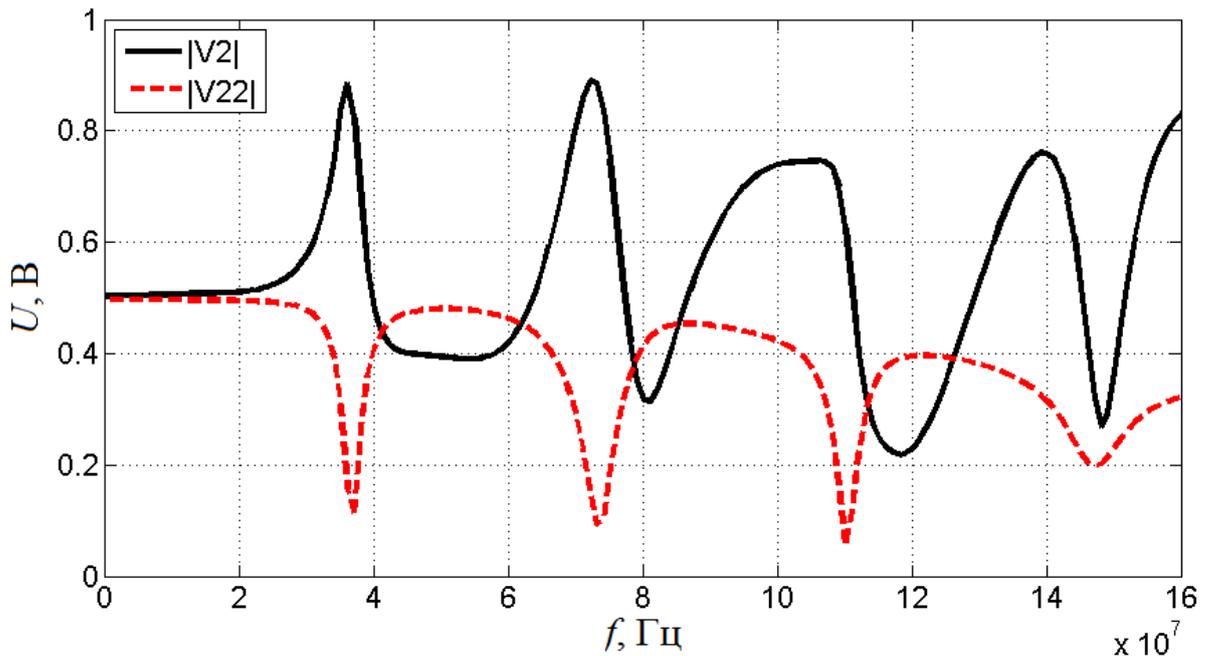


Рис. 4.20. АЧХ оптимального модального фильтра, полученного ЭС с ограничениями

Как и следовало ожидать, АЧХ имеет резонансы до 100 МГц. Таким образом, для решения поставленной задачи оптимизации необходимо

варьировать дополнительные параметры конструкции модального фильтра, например длину линий передачи. Однако оптимизация с помощью ЭС позволила получить меньшее значение максимальной амплитуды напряжения в узле 22 (5,21 В вместо 6,02 В в случае оптимизации перебором). При этом было использовано только 644 поколения и проанализировано 7728 конструкций модального фильтра, что в 2,1 раза меньше, чем в случае оптимизации перебором.

Заключение

В работе предложена новая эволюционная стратегия – $(\lambda)_{\text{opt}}\text{-}\sigma\text{CA-ЭС}$, выполнен ее анализ для сферической модели целевой функции, получена формула для расчета параметра самоадаптации τ . Тестирование программной реализации данной ЭС показало, что она сходится к минимуму сферической модели целевой функции с числом параметров 2–100 в 1,1–3,8 раза (в среднем по 100 независимым запускам) быстрее по сравнению со стандартной ЭС. Точность полученной формулы для расчета τ и выигрыш по скорости увеличиваются с ростом количества параметров, что делает $(\lambda)_{\text{opt}}\text{-}\sigma\text{CA-ЭС}$ практически значимым алгоритмом для задач оптимизации ЭМС с большим количеством параметров.

Разработаны геометрические модели конструктивных элементов ПП и корпусов БА КА для квазистатического и электродинамического анализа ЭМС, которые обеспечивают представление исходных данных о геометрии ПП и корпусов БА КА в форме, пригодной для обработки алгоритмами анализа ЭМС. Описана процедура моделирования ЭМС БА КА, на вход которой подаются данные о геометрии, компонентах и материалах ПП блока БА КА. В конце выполнения процедуры вычисленные для ПП БА КА значения амплитуд токов и напряжений (для квазистатического анализа) или напряженности электрического поля (для электродинамического анализа) сравниваются с пороговыми. Анализируемая ПП удовлетворяет требованиям ЭМС в случае, если вычисленные значения не превышают пороговые.

Разработана система компьютерного моделирования и оптимизации ЭМС БА КА. С использованием методов

объектно-ориентированного программирования реализовано ядро системы, которое обеспечивает добавление новых математических моделей и алгоритмов оптимизации без перекомпиляции системы. Встроенный в ядро системы интерпретатор языка скриптов с оригинальным набором специализированных команд позволяет создавать математические модели конструктивных элементов ПП и корпусов УЭМ БА КА. Клиенты системы обеспечивают работу пользователя через командную строку и графический интерфейс.

Проведено сравнение результатов натурального и вычислительного экспериментов на примере связанных линий печатной платы УЭМ БА КА: форма вычисленных в системе моделирования ЭМС БА КА сигналов согласуется с результатами натурального эксперимента, различие амплитуд составляет менее 15 %. С помощью системы компьютерного моделирования ЭМС БА КА выполнен анализ одиночной микрополосковой линии и дифференциальной пары, для которых получены контурные графики зависимости волнового сопротивления от геометрических и электрических параметров.

Осуществлена программная реализация модуля ЭС, в рамках которой разработаны алгоритмы выполнения скриптовых команд и вычисления целевой функции. Помимо стандартной ЭС и $(\lambda)_{opt}$ - σ СА-ЭС, реализованы алгоритмы $(\mu/\mu_W, \lambda)$ -АКМ-ЭС, $(\mu/\mu_I, \lambda)$ -АКМ- σ СА-ЭС и НФ, тестирование которых для сферической модели целевой функции показало корректность программной реализации.

Разработан алгоритм параллельного вычисления целевой функции, корректность которого обеспечивается реентерабельностью ядра. Выполнено тестирование алгоритма на задачах оптимизации параметров полосковой линии и со-

средоточенных нагрузок проводной структуры: для двух потоков выполнения получено уменьшение времени оптимизации соответственно в 1,85 и 1,75 раза (в среднем по 10 независимым запускам) по сравнению с однопоточным алгоритмом.

Предложена методика оптимизации ЭМС БА КА, с помощью которой получена оптимальная конфигурация полосковой линии (за 45 вычислений целевой функции разность емкостной и индуктивной связей для ППЛ и ОПЛ составила менее 10^{-3}) и оптимальные параметры нагрузок широкодиапазонной быстроразвращиваемой модифицированной вибраторной антенны (за 900 вычислений целевой функции максимальный КСВ в диапазоне 10–30 МГц снижен с 11,4 до 6,36). Для данных задач выполнено сравнение производительности алгоритма НФ, ГА, $(\mu/\mu_I, \lambda)$ - σ СА-ЭС, $(\lambda)_{opt}$ - σ СА-ЭС и $(\mu/\mu_W, \lambda)$ -АКМ-ЭС, которое показало, что для оптимизации параметров полосковой линии целесообразно использовать алгоритм НФ, а для оптимизации параметров антенной структуры – $(\mu/\mu_W, \lambda)$ -АКМ-ЭС. Для первой задачи с помощью ЭС автоматически выбран набор параметров алгоритма НФ, который при одинаковом бюджете вычислений целевой функции обеспечил более высокую точность решения по сравнению с исходными параметрами НФ.

Выполнена оптимизация параметров широкополосной математической модели низкочастотного резистора. За 10^4 вычислений целевой функции получено среднеквадратичное отклонение $2,23 \cdot 10^{-4}$ от экспериментального модуля коэффициента отражения в диапазоне до 20 ГГц.

Выполнена оптимизация конфигурации многопроводной линии передачи с числом проводников от 3 до 9, требуемый

характеристический импеданс с точностью лучше 1 Ом получен менее чем за $1,2 \cdot 10^4$ вычислений целевой функции. Рассмотрена задача структурной и параметрической оптимизации многокаскадного модального фильтра по двум критериям, синтезированы модальные фильтры с улучшенными характеристиками по сравнению с фильтрами, предложенными на основе экспертной оценки.

Созданная методика позволяет с помощью разработанной программной системы выбирать алгоритмы, обеспечивающие оптимизацию задач за минимальное количество вычислений целевой функции.

Результаты, изложенные в монографии, внедрены на ФГУП «ЦентрИнформ» и ОАО «Информационные спутниковые системы» имени академика М.Ф. Решетнева, а также в учебный процесс Томского государственного университета и Томского университета систем управления и радиоэлектроники.

Литература

1. Чермошенцев Ф. Автоматизация проектирования печатных плат цифровых электронных средств с учетом электромагнитной совместимости: дис. ... д-ра техн. наук / Ф. Чермошенцев. – Казань, 2004. – 474 с.
2. Heise E.R. Some simple spacecraft considerations / E.R. Heise, R.E. Heise // Proc. of IEEE EMC Symp. – 2006. – Vol. 1. – P. 182–186.
3. Калимулин И.Ф. Импеданс низкочастотных пассивных компонентов бортовой аппаратуры в диапазоне до 20 ГГц / И.Ф. Калимулин, Т.Р. Газизов, А.М. Заболоцкий // Приборы и техника эксперимента. – 2012. – № 2. – С. 91–97.
4. MIL-STD-461F. Requirements for the control of electromagnetic interference characteristics of subsystems and equipment. – 2007.
5. Electromagnetic compatibility requirements for space equipment and systems // Aerospace report TOR-2005(8583)-1. – 2005.
6. Иванов В.А. Модельные и стендовые исследования электризации космических аппаратов / В.А. Иванов, В.Ю. Кириллов, Е.П. Морозов; под ред. В.Ю. Кириллова. – М.: Изд-во МАИ, 2012. – 168 с.
7. High repetition rate picosecond FID pulse generators for UWB applications / V. Efanov [et al.] // Book of abstracts EUROEM 2012, 2–6 July 2012, Toulouse, France.
8. Paul C.R. Introduction to Electromagnetic Compatibility / C.R. Paul // Wiley Interscience. – 2006. – 836 p.
9. Perez R.P. Editorial Special Issue on Aerospace Electromagnetic Compatibility / R.P. Perez, J.A. Lukash //

Electromagnetic Compatibility, IEEE Transactions on. – 2008. – Vol. 50(3). – С. 453–454.

10. Газизов Т.Р. Искажения в межсоединениях и электромагнитный терроризм / Т.Р. Газизов // LAP Lambert Academic Publishing. – 2011. – 157 p.

11. Beyer H.-G. The Theory of Evolution Strategies / H.-G. Beyer // Natural Computing Series. – Springer, Heidelberg, 2001.

12. Arnold D.V. Noisy Optimization with Evolution Strategies / D.V. Arnold. – Kluwer Academic Publishers, Dordrecht, 2002.

13. Real-parameter black-box optimization benchmarking 2010: Experimental setup / N. Hansen [et al.] // GECCO (Companion). – 2010.

14. Comparing results of 31 algorithms from the black-box optimization benchmarking BBOB-2009 / N. Hansen [et al.] // GECCO (Companion). – 2010. – P. 1689–1696.

15. Газизов Т.Т. Алгоритмическое и программное обеспечение для моделирования проводных антенн с сосредоточенными нагрузками: дис. ... канд. техн. наук / Т.Т. Газизов. – Томск: Томск. гос. ун-т систем упр. и радиоэлектроники, 2008.

16. Kelley C.T. Iterative Methods for Optimization / C.T. Kelley // SIAM. – Philadelphia, 1999.

17. Finck S. Noisy optimization: a theoretical strategy comparison of ES, EGS, SPSA & if on the noisy sphere / S. Finck, H.-G. Beyer, A. Melkozerov // GECCO. – 2011. – P. 813–820.

18. Давыдов А.Г. Возможности программы ЭДЭМ для разработки устройств антенной техники / А.Г. Давыдов, Ю.В. Пименов // Антенны. – 2006. – № 12. – С. 54–66.

19. Тараканов В.П. Теоретический и численный анализ нелинейных задач физики плазмы посредством кода КАРАТ: дис. ... д-ра физ.-мат. наук: / В.П. Тараканов. – М., 2011. – 118 с.

20. Сайт компании-разработчика программного продукта CONCEPT-II [Электронный ресурс]. – Режим доступа: <http://www.tet.tu-harburg.de158>

21. Сайт компании-разработчика программного продукта FEKO [Электронный ресурс]. – Режим доступа: <http://www.feko.info/references/>

22. Сайт компании-разработчика программного продукта EMC Studio [Электронный ресурс]. – Режим доступа: <http://www.emcos.com/EMC>

23. Сайт компании-разработчика программного продукта WIPL-D [Электронный ресурс]. – Режим доступа: <http://www.wipl-d.com/products/products.php>

24. Сайт компании-разработчика программного продукта SINGULA [Электронный ресурс]. – Режим доступа: <http://www.integratedsoft.com>

25. Максимович Е.С. Использование программы электромагнитного моделирования IE3D для разработки планарных антенн / Е.С. Максимович // EDA Express. – 2005. – № 12. – С. 2–7.

26. Сайт компании-разработчика программного продукта SEMCAD-X [Электронный ресурс]. – Режим доступа: <http://www.semcad.com>

27. Сайт компании-разработчика программного продукта EMPIRE [Электронный ресурс]. – Режим доступа: <http://www.empire.de>

28. Сайт компании-разработчика программного продукта XFDTD [Электронный ресурс]. – Режим доступа: <http://www.remcom.com/>

29. Сайт компании-разработчика программного продукта GEMS [Электронный ресурс]. – Режим доступа: <http://www.2comu.com/>

30. Сайт компании-разработчика программного продукта FIDELITY [Электронный ресурс]. – Режим доступа: <http://www.systemfidelity.com/>

31. Сайт компании-разработчика программного продукта CSTMICROSTRIPES [Электронный ресурс]. – Режим доступа: <http://www.cst.com/159>

32. Сайт компании-разработчика программного продукта HFSS [Электронный ресурс]. – Режим доступа: <http://www.ansys.com/>

33. Сайт компании-разработчика программного продукта CSTMICROWAVE STUDIO [Электронный ресурс]. – Режим доступа: <http://www.cst.com/>

34. Разработка и поставка аппаратно-программного комплекса для проведения анализа взаимовлияний электрических сигналов бортовой аппаратуры: отчет об ОКР (промежуточ.) / Томск. гос. ун-т систем упр. и радиоэлектроники. – Томск, 2009. – Док. № 28/08 от 14.04.2008, шифр «АПК–ТУСУР».

35. Beyer H.-G. Evolution strategies / H.-G. Beyer // Scholarpedia. – 2007. – № 2(7). – P. 1965.

36. Beyer H.-G. Evolution Strategies: A Comprehensive Introduction / H.-G. Beyer, H.-P. Schwefel // Natural Computing. – 2002. – № 1(1). – P. 3–52.

37. Experimentelle Optimierung einer zweiphasigen: see teil. technical report no. 35 of the project mhd-staustahlrohr 11.034/68 / AEG Research Institute. – Berlin; H.-P. Schwefel, 1968.

38. Rechenberg I. Evolutionsstrategie: Optimierung technischer Systemenach Prinzipien der biologischen Evolution: PhD thesis, Dr.-Ing. / I. Rechenberg // Thesis, Technical University of Berlin: Department of Process Engineering. – 1971.

39. Rechenberg I. Evolutionsstrategie: Optimierung technischer Systemenach Prinzipien der biologischen Evolution / I. Rechenberg. – Stuttgart: Frommann-Holzboog Verlag, 1973.

40. Schwefel H -P. Evolutionsstrategie und numerische Optimierung: Dissertation / H.-P. Schwefel. – TU Berlin, Germany, 1975.

41. Schwefel H.-P. Numerical Optimization of Computer Models / H.-P. Schwefel. – Wiley, Chichester, 1981.

42. Rastrigin L.A. Random search in optimization problems for multiparametersystems (translated from the Russian original: Sluchainyi poisk v zadachakh 160 optimisatsii mnogoparametricheskikh sistem, Zinatne, Riga) / L.A. Rastrigin // Air Force System Command Foreign Technology Division, FTD-HT-67-363. – 1965.

43. Фогель Л. Искусственный интеллект и эволюционное моделирование / Л. Фогель, А. Оуэнс, М. Уолш. – М.: Мир, 1969.

44. Ивахненко А.Г. Предсказание случайных процессов / А.Г. Ивахненко, В.Г. Лапа. – Киев: Наукова думка, 1971.

45. Букатова Л.И. Эволюционное моделирование: идеи, основы теории, приложения / Л.И. Букатова. – М.: Знание, 1981.

46. Семенов М.А. О подходе к доказательству сходимости эволюционных методов / М.А. Семенов, Д.А. Теркель // Перспективы развития вычислительных систем. – Рига: РПИ, 1985. – С. 92–102.

47. Ostermeier A. Step-size adaptation based on non-local use of selection information / A. Ostermeier, A. Gawelczyk, N. Hansen // Parallel Problem Solving from Nature—PPSN III. – 1994. – P. 189–198.

48. Suttorp Thorsten. Efficient covariance matrix update for variable metric evolution strategies / Thorsten Suttorp, Nikolaus Hansen, Christian Igel // Machine Learning. – 2009. – № 75(2). – P. 167–197.

49. Igel C. Covariance matrix adaptation for multiobjective optimization / C. Igel, N. Hansen, S. Roth // Evolutionary Computation. – 2007. – № 15(1). – P. 1–28.

50. Glass C.W. Uspeh — evolutionary crystal structure prediction / C.W. Glass, A.R. Oganov, N. Hansen // Comp. Phys. Comm. – 2006. – № 175. – P. 713–720.

51. Meyer-Nieberg S. Self-Adaptation in Evolution Strategies / S. Meyer-Nieberg PhD thesis, University of Dortmund, CS Department. – Dortmund, Germany, 2007.

52. Meyer-Nieberg S. Self-Adaptation in Evolutionary Algorithms / S. Meyer-Nieberg, H.-G. Beyer; F.G. Lobo, C.F. Lima, Z. Michalewicz, editors // Parameter Setting in Evolutionary Algorithms. – Springer, Berlin, 2007. – P. 47–75.

53. Finck S. Analysis of Evolution Strategies on a Subset of Quadratic Functions and Methods for Comparing Optimization Strategies / S. Finck // PhD thesis, University of Stuttgart. – Stuttgart, Germany, 2011.

54. Arnold D.V. Evolution Strategies with Cumulative Step Length Adaptation on the Noisy Parabolic Ridge / D.V. Arnold, H.-G. Beyer // Natural Computing. – 2006. – accepted.

55. Oyman A.I. Analysis of a Simple ESON on the «Parabolic Ridge» / A.I. Oyman, H.-G. Beyer, H.-P. Schwefel // Technical Report SyS-2/97 / University of Dortmund, Department of Computer Science, Systems Analysis Research Group. – 1997, August.

56. Mutative Self-Adaptation on the Sharp and Parabolic Ridge / S. Meyer-Nieberg [et al.] // Foundations of Genetic Algorithms. – Berlin; Springer-Verlag, 2007. – № 9. – P. 70–96.

57. Katok A. Introduction to the modern theory of dynamical systems / A. Katok, B. Hasselblatt // Encyclopedia of mathematics and its applications. – Cambridge University Press, 1996.

58. Hansen N. Completely derandomized self-adaptation in evolution strategies / N. Hansen, A. Ostermeier // Evolutionary computation. – 2001. – № 9(2). – P. 159–195.

59. A method for handling uncertainty in evolutionary optimization with an application to feedback control of combustion / N. Hansen [et al.] // IEEE Transactions on Evolutionary Computation. – 2009. – № 13(1). – P. 180–197.

60. Hansen N. Reducing the Time Complexity of the Derandomized Evolution Strategy with Covariance Matrix Adaptation (CMA-ES) / N. Hansen, S.D. Muller, P. Koumoutsakos // Evolutionary Computation. – 2003. – № 11(1). – P. 1–18.

61. Beyer H.G. Covariance Matrix Adaptation Revisited – The 162 CMSA Evolution Strategy / H.G. Beyer, B. Sendhoff // Parallel Problem Solving from Nature. – 2008. – P. 123–132.

62. Eiben A.E. Introduction to Evolutionary Computing / A.E. Eiben, J.E. Smith // Natural Computing Series. – Springer, Berlin, 2003.

63. Kramer O. A Review of Constraint-Handling Techniques for Evolution Strategies / O. Kramer // Applied Computational Intelligence and Soft Computing. – 2010. – Article ID 185063. – 11 p.

64. Mezura-Montes E. A simple multimembered evolutionstrategy to solve constrained optimization problems / E. Mezura-Montes, C.A.C. Coello // Evolutionary Computation, IEEE Transactions on. – 2005. – № 9(1). – P. 1–17.

65. Beyer H.G. On the design of constraint covariance matrix self-adaptation evolution strategies including a cardinality constraint / H.G. Beyer, S. Finck // Evolutionary Computation, IEEE Transactions on. – 2011. – № 99. – P. 1.

66. Елшин Ю. Как создать отечественную САПР? / Ю. Елшин // Стрела. – 2007, ноябрь. – С. 2.

67. Дрот В.Л. Толковый словарь современной компьютерной лексики / В.Л. Дрот, Ф.А. Новиков. – 3-е изд., перераб. и доп. – СПб.: БХВ-Петербург, 2004. – 608 с.

68. Arnold D.V. Weighted multirecombination evolution strategies / D.V. Arnold // Theoretical computer science. – 2006. – № 361(1). – P. 18–37.

69. Arnold D.V. On the behaviour of weighted multi-recombination evolution strategies optimising noisy cigar functions / D.V. Arnold, H.-G. Beyer, A. Melkozerov // GECCO. – 2009. – P. 483–490.

70. TALGAT 2008: свидетельство о государственной регистрации программы для ЭВМ № 2009614871 / Т.Р. Газизов, А.О. Мелкозеров, Т.Т. Газизов, С.П. Куксенко, А.М. Заболоцкий. – Зарег. 08.09.09.

71. TALGAT 2009: свидетельство о государственной регистрации программы для ЭВМ № 2010613497 / Т.Р. Газизов, А.О. Мелкозеров, Т.Т. Газизов, С.П. Куксенко, А.М. Заболоцкий. – Зарег. 28.05.10.

72. TALGAT 2010: свидетельство о государственной регистрации программы для ЭВМ № 2012610712 / Т.Р. Газизов, А.О. Мелкозеров, Т.Т. Газизов, С.П. Куксенко, А.М. Заболоцкий, Р.И. Аширбакиев, Е.А. Вершинин, В.К. Салов, Е.В. Лежнин. – Зарег. 13.01.12.

73. Система компьютерного моделирования сложных структур проводников и диэлектриков TALGAT: свидетельство об отраслевой регистрации разработки / Т.Р. Газизов, А.О. Мелкозеров, Т.Т. Газизов, С.П. Куксенко, А.М. Заболоцкий, И.С. Костарев. – № 8376 от 24.05.07. – Зарег. в Отраслевом фонде алгоритмов и программ Госкоорцентра Минобрнауки РФ с присвоением номера государственной регистрации – Рег. № ВНТИЦ 50200701103.

74. Система компьютерного моделирования сложных структур проводников и диэлектриков / Т.Р. Газизов [и др.] // Компьютерные учебные программы и инновации. – М.: ГОСКООРЦЕНТР, МФЮА, РУИ. – 2007. – № 10. – С. 89–90.

75. Мелкозеров А.О. Реализация модуля эволюционных стратегий / А.О. Мелкозеров // Материалы докл. всерос. науч.-техн. конф. студентов, аспирантов и молодых ученых «Научная сессия ТУСУР-2010». – Томск, 2010. – Ч. 1. – С. 160–163.

76. Компьютерное моделирование сложных структур проводников при проектировании телевизионно-вычислительных систем / Т.Р. Газизов [и др.] // Известия вузов. Приборостроение. – 2005. – № 10. – С. 63–66.

77. Мелкозеров А.О. Применение методов обобщенного программирования в разработке системы компьютерного моделирования сложных структур проводников и диэлектриков / А.О. Мелкозеров // Материалы науч.-техн. конф. студентов и молодых ученых «Научная сессия ТУСУР-2003», 13–15 мая 2003 г. – Томск, 2003. – В 3 ч. – Ч. 1. – С. 117–120.

78. Пат. 66613 Российская Федерация. Широкополосная антенная система для работы в декаметровом диапазоне / Н.Д. Малютин, Т.Т. Газизов, Т.Р. Газизов, А.О. Мелкозеров, А.Г. Лоцилов, Э.В. Семенов, А.П. Рыбин. – № 2007114313; опубл. 10.09.07, Бюл. № 25; приоритет 16.04.07.

79. Пат. 79355 Российская Федерация. Модальный фильтр / Т.Р. Газизов, И.Г. Бевзенко, А.О. Мелкозеров, И.Е. Самотин, П.Е. Орлов, А.М. Заболоцкий, Т.Т. Газизов, С.П. Куксенко, И.С. Костарев. – № 2008127527/22(033781); опубл. 27.12.08, Бюл. № 36; приоритет 07.07.2008.

80. Пат. 79355 Российская Федерация. Устройство воздействия на аппаратуру / Т.Р. Газизов, И.Г. Бевзенко, А.О. Мелкозеров, И.Е. Самотин, П.Е. Орлов, А.М. Заболоцкий, Т.Т. Газизов, С.П. Куксенко, И.С. Костарев. – № 2008127527/22(033781); опубл. 27.12.08, Бюл. № 36; приоритет 07.07.08.

81. Пат. 800100 Российская Федерация. Устройство модального зондирования / Т.Р. Газизов, И.Г. Бевзенко, А.О. Мелкозеров, И.Е. Самотин, П.Е. Орлов, А.М. Заболоцкий, Т.Т. Газизов, С.П. Куксенко, И.С. Костарев. – № 2008127580/22(033837); опубл. 20.01.2009, Бюл. № 2; приоритет 07.07.08.

82. Пат. 2386964 Российская Федерация. Устройство обна-ружения, идентификации и диагностики многопроводных линий передачи / Т.Р. Газизов, И.Г. Бевзенко, А.О. Мелкозеров, И.Е. Самотин, П.Е. Орлов, А.М. Заболоцкий, Т.Т. Газизов, С.П. Куксенко, И.С. Костарев. – № 2009108905/28(011919); опубл. 20.04.10, Бюл. № 11; приоритет 10.03.09.

83. Пат. 431912 Российская Федерация. Устройство для нарушения работы аппаратуры за счет разложения и восстановления импульсов / Т.Р. Газизов, И.Г. Бевзенко, А.О. Мелкозеров, И.Е. Самотин, П.Е. Орлов, А.М. Заболоцкий, Т.Т. Газизов, С.П. Куксенко, И.С. Костарев. – № 2010108518/07(012013); опубл. 20.10.11, Бюл. № 29; приоритет 09.03.10.

84. Пат. 2431897 Российская Федерация. Устройство защиты от импульсных сигналов / Т.Р. Газизов, И.Г. Бевзенко, А.О. Мелкозеров, И.Е. Самотин, П.Е. Орлов, А.М. Заболоцкий, Т.Т. Газизов, С.П. Куксенко, И.С. Костарев. – № 2010108520/07(012016); опубл. 20.10.11, Бюл. № 29; приоритет 09.03.10.

85. Evaluations of Protection Methods Using TVSarray and Modal Filter / T.R. Gazizov [et al.] // Book of abstracts EUROEM 2012, 2–6 July 2012. – Toulouse, France. – P. 106.

86. Разработка комплекса программных и технических средств для контроля информационных магистралей, обеспечения электромагнитной совместимости и исследования надежности унифицированного ряда электронных модулей на основе технологии «система-на-кристалле» для систем управления и электропитания космических аппаратов связи, навигации и дистанционного зондирования Земли с длительным сроком активного существования: отчет об ОКР (промежуточ.) / Томск. гос. ун-т систем упр. и радиоэлектроники. – Томск, 2012. – х/д 95/10 от 24.11.2010.

87. Ускорение вычислений в задачах моделирования ЭМС / В.К. Салов [и др.] // Тр. 9-го междунар. симп. по электромагнитной совместимости и электромагнитной экологии, г. Санкт-Петербург, 13–16 сентября 2011 г. – СПб., 2011. – С. 265–268.

88. Мелкозеров А.О. Исследование точности вычисления емкостных матриц конфигурации из двух проводов малого диаметра / А.О. Мелкозеров, Т.Р. Газизов // Материалы всерос. науч.-техн. конф. молодых ученых и студентов «Современные проблемы радиоэлектроники», 6–7 мая 2004 г. – Красноярск, 2004. – С. 669–672.

89. Газизов Т.Р. Квазистатический анализ многопроводных электрических соединений в системе TALGAT / Т.Р. Газизов, А.О. Мелкозеров, Р.И. Аширбакиев // 9-й междунар. симп. по электромагнитной совместимости и электромагнитной экологии: сб. тр., Санкт-Петербург, 13–16 сентября 2011 г. – СПб., 2011. – С. 265–268.

90. Александреску А. Современное проектирование на C++: пер. с англ. / А. Александреску // Сер. C++ In-Depth. – М.: Вильямс, 2002. – Т. 3.

91. Саттер Г. Решение сложных задач на C++: пер. с англ. / Г. Саттер // Сер. C++ In-Depth. – М.: Вильямс, 2002. – Т. 4.

92. Приемы объектно-ориентированного программирования. Паттерны проектирования / Э. Гамма [и др.] // Сер. Библиотека программиста. – СПб.: Питер, 2001.

93. Лежнин Е.В. Улучшение пользовательского интерфейса системы TALGAT / Е.В. Лежнин, А.О. Мелкозеров // Доклады ТУСУР. – 2011. – № 2(24). – Ч. 1. – С. 137–140.

94. Лежнин Е.В. Редактор двумерных конфигураций проводников и диэлектриков в системе TALGAT / Е.В. Лежнин, А.О. Мелкозеров // Материалы докл. всерос. науч.-техн. конф. студентов, аспирантов и молодых ученых «Научная сессия ТУСУР-2012». – Томск, 2012. – С. 106–109.

95. Аширбакиев Р.И. Реализация модуля импорта геометрических и электрических параметров из формата Altium Designer/Protel ASCII в формат системы TALGAT / Р.И. Аширбакиев, Е.В. Лежнин, А.О. Мелкозеров // Материалы докл. всерос. науч.-техн. конф. студентов, аспирантов и молодых ученых «Научная сессия ТУСУР-2012». – Томск, 2012. – С. 72–75.

96. Суровцев Р.С. Выбор оптимальных параметров дифференциальной пары на печатной плате / Р.С. Суровцев, А.О. Мелкозеров // Материалы докл. всерос. науч.-техн. конф. студентов, аспирантов и молодых ученых «Научная сессия ТУСУР-2012». – Томск, 2012. – С. 123–126.

97. Соловьев А.В. Анализ современных методов расчета электрофизических параметров печатных плат и их соответствия современным требованиям при разработке быстродействующей электроники / А.В. Соловьев // 7-й междунар. симп. по электромагнитной совместимости и электромагнитной экологии: сб. тр., Санкт-Петербург, 26–29 июня 2007 г. – СПб., 2007. – С. 301–303.

98. Beyer H.-G. Sigma-Self-Adaptive Weighted Multirecombination Evolution Strategy with Scaled Weights on the Noisy Sphere / H.-G. Beyer, A. Melkozerov // PPSN. – 2008. – P. 11–20.

99. Beyer H.-G. Mutative sigma-self-adaptation can beat cumulative step size adaptation when using weighted recombination / H.-G. Beyer, A. Melkozerov // GECCO. – 2008. – P. 487–494.

100. Hansen N. Completely derandomized self-adaptation in evolution strategies / N. Hansen, A. Ostermeier // Evolutionary Computation. – 2001. – № 9(2). – P. 159–195.

101. Meyer-Nieberg S. On the analysis of self-adaptive recombination strategies: first results / S. Meyer-Nieberg, H.-G. Beyer // Congress on Evolutionary Computation. – 2005. – P. 2341–2348.

102. Local performance of the $(\mu/\mu_i, \lambda)$ -es in a noisy environment / H.-G. Beyer [et al.] // Foundations of Genetic Algorithms. – San Francisco, CA, 2001. – № 6. – P. 127–141.

103. Мелкозеров А.О. Реализация генетического алгоритма в системе компьютерного моделирования сложных структур проводников и диэлектриков / А.О. Мелкозеров // Материалы всерос. науч.-техн. конф. «Научная сессия ТУСУР-2004». –Томск, 2004. – В 3 ч. – Ч. 1. – С. 105–107.

104. Мелкозеров А.О. Структурно-параметрическая оптимизация генетическими алгоритмами / А.О. Мелкозеров, Т.Р. Газизов // Материалы всерос. науч.-практ. конф. «Проблемы информационной безопасности государства, общества и личности», 2–4 июня 2004 г. – Томск, 2004. – С. 122–124.

105. Мелкозеров А.О. Структурная оптимизация генетическими алгоритмами системы полуволновых диполей / А.О. Мелкозеров, Т.Р. Газизов // Материалы всерос. науч.-практ. конф. «Проблемы информационной безопасности государства, общества и личности», 16–18 февраля 2005 г. – Томск, 2005. – С. 57–61.

106. Yieldoptimization using a GaAs process simulator coupled to a physical devicemodel / D.E. Stoneking [et al.] // In Proceedings IEEE/Cornell Conference on Advanced Concepts in High Speed Devices and Circuits, IEEE – 1991. – P. 374–383.

107. Мелкозеров А.О. Исследование точности нахождения оптимальных параметров полосковых линий с помощью генетических алгоритмов / А.О. Мелкозеров // Материалы всерос. науч.-техн. конф. студентов и молодых ученых «Научная сессия ТУСУР-2005». – Томск, 2005. – Ч. 1. – С. 126–129.

108. Газизов Т.Р. Уменьшение искажений электрических сигналов в межсоединениях / Т.Р. Газизов. – Томск: НТЛ, 2003. – 169 с.

109. Газизов Т.Т. Электродинамическое моделирование произвольных проводных структур / Т.Т. Газизов, Т.Р. Газизов, А.О. Мелкозеров // Материалы всерос. науч.-практ. конф. «Проблемы информационной безопасности государства, общества и личности», 16–18 февраля 2005 г. – Томск, 2005. – С. 47–51.

110. Широкодиапазонные приемопередающие комбинированные антенны. Принципы построения. Решение внутренней задачи / Н.Д. Малютин [и др.] // Материалы междунар. науч.-практ. конф. «Электронные средства и системы управления», г. Томск, 6–8 октября 2004 г. – Томск: Изд-во Ин-та оптики атмосферы СО РАН, 2004. – С. 107–111.

111. Газизов Т.Т. Автоматизированное проектирование оптимальных антенн с сосредоточенными нагрузками / Т.Т. Газизов, А.О. Мелкозеров // Материалы междунар. науч.-практ. конф. «Электронные средства и системы управления», 12–14 октября 2005 г. – Томск, 2005. – С. 149–152.

112. NEC based antenna modeler and optimizer. – URL: <http://home.ict.nl/arivoors/>

113. Altman Z. New designs of ultra wide-band communication antennas using a genetic algorithm / Z. Altman, R. Mittra, A. Boag // IEEE Antennas and Propagation Magazine. – 1997. – Vol. 45. P. 1494–1501.

114. A method to construct equivalent circuit model from frequency responses with guaranteed passivity / I.R. Pordanjani [et al.] // Power Delivery, IEEE Transactions on. – 2011. – № 26(1). – P. 400–409.

115. Inc. Sonnet Software. Planar EM Analysis: A New Standard for High Frequency Applications // Microwave Journal. – № 47(11). – P. 140–149, 204.

116. Fonseca Carlos M. An overview of evolutionary algorithms in multiobjective optimization / Carlos M. Fonseca, Peter J. Fleming // *Evol. Comput.* – 1995. – № 3. – P. 1–16.

117. Coello C.A.C. An updated survey of evolutionary multiobjective optimization techniques: State of the art and future trends / C.A.C. Coello // *Evolutionary Computation*, 1999. CEC 99. Proceedings of the 1999 Congress on. IEEE. – 1999. – Vol. 1.

118. Altinoz O.T. Particle swarm optimization with parameter dependency walls and its sample application to the microstrip-like inter connect line design / O.T. Altinoz, A.E. Yilmaz // *AEU-International Journal of Electronics and Communications*. – 2011. – P. 107–114.

119. Rahmat-Samii Y. Guest editorial for the special issue on synthesis and optimization techniques in electromagnetics and antenna system design / Y. Rahmat-Samii, C.G. Christodoulou // *Antennas and Propagation, IEEE Transactions on*. – 2007. – № 55(3). – P. 518–522.

120. Real-parameter black-box optimization benchmarking 2010: Experimental setup / N. Hansen, A. Auger, S. Finck, R. Ros // *Technical Report RR-7215, INRIA*. – 2010.

121. Структурная оптимизация многокаскадного модального фильтра по двум критериям / А.О. Мелкозеров [и др.] // *Доклады ТУСУР*. – 2010. – № 2(22), ч. 1. – С. 70–72.

122. Analysis of power dissipation in resistive terminations of single- and multistage modal filters / T.R. Gazizov [et al.] // *Proc. of 31-th Int. conf. on lightning protection*. – Vienna, Austria.

123. Improved design of modal filter for electronics protection / T.R. Gazizov [et al.] // *Proc. of 31-th Int. conf. on lightning protection*. – Vienna, Austria.

124. Кечиев Л.Н. Проектирование печатных плат для цифровой быстродействующей аппаратуры / Л.Н. Кечиев. – М.: Группа ИДТ, 2007. – 616 с.

125. Мелкозеров А.О. Методика оптимизации электромагнитной совместимости бортовой аппаратуры перспективных космических аппаратов / А.О. Мелкозеров // Известия высших учебных заведений. Физика. – 2012. – Ч. 9-3. – С. 91–95.

Оглавление

Введение	5
1. Анализ методов моделирования и оптимизации электромагнитной совместимости	9
1.1. Актуальность проблемы электромагнитной совместимости	9
1.2. Методы и системы электродинамического моделирования	13
1.3. Эволюционные стратегии	27
1.4. Обоснование необходимости создания собственной системы моделирования и оптимизации	42
2. Алгоритмы и модели для моделирования и оптимизации электромагнитной совместимости бортовой аппаратуры космических аппаратов	49
2.1. Алгоритм эволюционной стратегии со взвешенной мультирекомбинацией и самоадаптацией интенсивности мутации	49
2.2. Теоретический анализ эволюционных стратегий	53
2.3. Анализ мультирекомбинационной эволюционной стратегии	55
2.4. Анализ $(\mu/\mu_I, \lambda)$ - σ СА эволюционной стратегии	60
2.5. Пространства поиска конечной размерности	64
2.6. Сравнительный анализ $(\mu/\mu_I, \lambda)$ и $(\lambda)_{\text{opt}}$ эволюционных стратегий	67

2.7.	Разработка геометрических моделей конструктивных элементов печатных плат и корпусов бортовой аппаратуры космических аппаратов для квазистатического и электродинамического анализа электромагнитной совместимости	71
2.8.	Моделирование электромагнитной совместимости бортовой аппаратуры космических аппаратов	75
3.	Программная реализация системы компьютерного моделирования и оптимизации электромагнитной совместимости бортовой аппаратуры космических аппаратов	79
3.1.	Структура системы TALGAT	79
3.2.	Ядро системы TALGAT	81
3.3.	Язык скриптов	85
3.4.	Динамические модули	94
3.5.	Клиенты системы TALGAT	97
3.6.	Тестирование системы TALGAT	101
3.7.	Программная реализация модуля эволюционных стратегий	128
3.8.	Программная реализация алгоритма неявного фильтрования	142
4.	Методика оптимизации электромагнитной совместимости бортовой аппаратуры космических аппаратов	148
4.1.	Описание методики оптимизации	148
4.2.	Оптимизация параметров полосковых линий	152

4.3.	Оптимизация параметров проводных структур	164
4.4.	Оптимизация параметров математической модели резистора	171
4.5.	Оптимизация параметров многопроводной линии передачи	177
4.6.	Структурная и параметрическая оптимизация многокаскадного модального фильтра	188
	Заключение	195
	Литература	199

Научное издание

Мелкозеров Александр Олегович

Аширбакиев Ренат Ихсанович

**КОМПЬЮТЕРНОЕ МОДЕЛИРОВАНИЕ И ОПТИМИЗАЦИЯ
ЭЛЕКТРОМАГНИТНОЙ СОВМЕСТИМОСТИ БОРТОВОЙ
АППАРАТУРЫ КОСМИЧЕСКИХ АППАРАТОВ**

Монография

Корректор Л.И. Кирпиченко

Компьютерная верстка Р.И. Аширбакиева

Подписано в печать 27.09.13. Формат 60x84/16. Усл.-печ. л. 12,79.

Тираж 500 экз. Заказ 931.

Томский государственный университет
систем управления и радиоэлектроники.

634050, г. Томск, пр. Ленина, 40.

Тел. (3822) 533018.