

Л.А. Растрюгин, Ю.П. Пономарев

ЭКСТРАПОЛЯЦИОННЫЕ МЕТОДЫ проектирования и управления

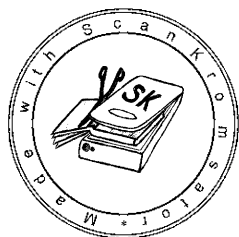
Л.А. Растрюгин, Ю.П. Пономарев

ЭКСТРАПОЛЯЦИОННЫЕ МЕТОДЫ проектирования и управления



МОСКВА «МАШИНОСТРОЕНИЕ» 1986

ББК 32.81
Р 24
УДК 519.7



Рецензент д-р техн. наук проф. Л.Г. Евланов

Растрингин Л.А., Пономарев Ю.П.

Р 24 Экстраполяционные методы проектирования и управления. — М.: Машиностроение, 1986. — 120 с., ил.
40 к.

Книга посвящена возникающим при проектировании и управлении сложными системами вопросам, которые связаны с прогнозированием ожидаемых параметров этих систем.

В основу положен предложенный и разрабатываемый авторами метод многомерной линейной экстраполяции (МЛЭ), позволяющий решать широкий круг задач проектирования и управления.

Многочисленные примеры иллюстрируют работу конкретных алгоритмов. Приведены оценки точности метода многомерной экстраполяции. Определены основные направления исследований и пути построения модифицированных алгоритмов многомерной экстраполяции.

Книга будет полезна специалистам, занимающимся вопросами проектирования сложных систем и их управлением.

Р 1502000000-005
038 (01)-86 5-86

ББК 32.81
6Ф0.1

ПРЕДИСЛОВИЕ

Пожалуй одной из самых привлекательных сторон деятельности человека является его умение принимать решения при очень малой исходной информации. При этом человек пользуется своим или чужим опытом принятия решения в этой области. Таким образом, этот опыт и сложившаяся ситуация позволяют ему эффективно принимать ответственные решения.

Естественно желание смоделировать этот процесс, не вдаваясь в его наверняка чрезвычайно сложные психологические механизмы.

Эвристический подход в этом случае является, по-видимому, наиболее плодотворным. Существующие эвристики (они описаны в гл. 1) не удовлетворяют всем требованиям, предъявляемым для процессов принятия сложных решений.

В данной книге описывается новый метод, использующий эвристику линейной экстраполяции (точнее: кусочно-линейной). Она опирается на очевидные соображения, что похожие ситуации должны вызывать похожие решения. При этом точность принимаемого решения должна зависеть от объема предыдущего опыта, но в любом случае давать решение даже, если опыт был однократным (ценой повышения ошибки, разумеется).

Многочисленные применения предложенного подхода в проектировании и оперативном управлении (некоторые из них приведены в книге) показали его высокую эффективность и достаточную надежность. Все это в сочетании с простотой метода позволяет рекомендовать его для оценочных расчетов, а иногда, когда объем предыдущего опыта велик, и для точных решений.

Книга состоит из четырех глав. В гл. I проводится анализ методов математического прогнозирования, формируются требования к методу многомерной экстраполяции и излагается суть этого метода. Знакомство с этим материалом не требует высокой математической подготовки, а необходимые сведения из линейной алгебры приведены в отдельном параграфе. Рассматривается направление повышения точности метода многомерной экстраполяции, основанное на использовании подвижных подпространств. В заключение главы анализируется точность метода многомерной линейной экстраполяции.

Гл. II посвящена вопросам применения метода многомерной линейной экстраполяции для решения задач проектирования. Основное внимание уделяется синтезу простейших алгоритмов, которые не требуют больших затрат машинного времени.

В гл. III рассматриваются вопросы применения метода многомерной линейной экстраполяции для решения задач оптимального управления.

В гл. IV описаны пути совершенствования предложенного метода. В этой главе использованы материалы, полученные В.А. Агладзе.

Авторы надеются, что материалы, изложенные в книге, окажут помощь широкому кругу специалистов, занимающихся в той или иной мере вопросами проектирования сложных систем и их управлением.

Введение

Наиболее верный путь повышения эффективности научных исследований, влияющих на технический прогресс, — сквозное планирование на всех этапах, начиная с появления научной идеи до ее практического воплощения [21].

Любая техническая система или устройство постоянно развивается и пополняется все более совершенными образцами. Смена одного образца техники другим позволяет и заставляет говорить о его жизненном цикле, являющемся упорядоченной совокупностью фаз или состояний, которые проходит этот образец с момента возникновения идеи о его создании до снятия с эксплуатации.

Если говорить о трудностях, которые приходится преодолевать при сквозном планировании создания образцов новой техники, то в первую очередь нужно остановиться на необходимости учета большого числа неопределенностей и среди них — неопределенностей в оценке истинных потребностей в образце новой техники, технических и экономических неопределенностей, связанных с оценкой путей создания образца с заданными техническими характеристиками, неопределенностей в оценке продолжительности отдельных фаз и стадий его жизненного цикла и затрат на их реализацию и т.д. Технические неопределенности порождаются неоднозначностью структуры образцов новой техники. Многовариантность структуры в свою очередь обусловлена множеством типов элементов и их параметров.

Большую роль при принятии решения о создании перспективного образца новой техники играет точность прогнозирования продолжительности реализационного периода, т.е. части жизненного цикла от формирования замысла до серийного производства данного образца. Если указанная продолжительность окажется чрезмерно заниженной, то это может привести к тому, что созданный образец новой техники поступит в эксплуатацию, когда он будет уже морально устаревшим. Объясняется это тем, что смена поколений образцов техники в передовых отраслях промышленности происходит в среднем в течение 5 — 7 лет [21]. Особенно трудно решать задачу прогноза продолжительности реализационного периода образца новой техники на начальных фазах его жизненного цикла (формирование замысла, проведение целевых НИР).

Таким образом, неопределенности в исходных данных — объективное свойство условий, в которых приходится принимать решение о создании того или иного образца новой техники. Причем эти неопределенности велики на начальных стадиях (назовем их стадиями предварительного проектирования) жизненного цикла и быстро убывают по мере перехода от одной фазы к другой. Отсюда следует важный вывод о том, что нецелесообразно решение о создании образца новой техники, если этот образец находится на одной из начальных фаз жизненного цикла. Управление процессом создания образца новой техники, с учетом присущих этому процессу неопределенностей, должно быть пофазным. Это означает, что

после завершения каждой фазы жизненного цикла принимается новое решение о продолжении работ по созданию образца новой техники, а все ранние решения теряют силу (если они не совпадают с последними или не включаются в него). Кроме того, после завершения каждой фазы вновь решается задача прогнозирования жизненного цикла, а также составляется план выполнения очередной фазы. Данное требование к управлению процессом создания образца новой техники следует из того, что завершение каждой очередной фазы жизненного цикла может сопровождаться получением данных, способных существенным образом изменить картину (вплоть до перевода создаваемого образца в разряд неэффективных).

Решение о продолжении выполнения работ после завершения каждой очередной фазы должно приниматься как минимум по следующим критериям: эффективности образца; вероятности создания образца с заданными техническими характеристиками; стоимости работ, которые необходимо выполнить на последующих фазах, чтобы можно было приступить к серийному производству образца; стоимости серийного производства одного вида образцов; времени поступления образца в эксплуатацию; продолжительности периода полезной жизни образца. В каждом конкретном случае перечень указанных критериев может быть расширен (влияние нового образца на перспективы развития других образцов техники, степень использования при создании образца самых последних достижений науки и техники во всех конструкторских и технологических решениях и др.). После разработки технических предложений (аванпроектов) из тех конкурирующих образцов новой техники, которые удовлетворяют всем рассматриваемым критериям, необходимо выбрать один образец по критерию "затраты — эффективность". По этому критерию предпочтение отдается тому образцу новой техники, который при одинаковых затратах имеет большую эффективность или при одинаковой эффективности требует меньших затрат.

Из сказанного следует, что основным методом управления процессом создания образцов новой техники должен явиться программно-целевой метод, который в последнее время находит все более широкое распространение. Этот метод предполагает принятие решения по сложным проблемам на основе заранее подготовленных комплексных программ, которые направлены на достижение одной или нескольких целей [21]. Применение программно-целевого метода влечет за собой необходимость выполнения следующих двух требований.

Во-первых, необходимо сквозное планирование процесса выполнения совокупности взаимосвязанных работ, направленных на создание образца новой техники, начиная с формирования замысла и кончая серийным производством. В этом случае весь цикл "исследование — производство" рассматривается как единый процесс. Сквозное планирование позволяет сократить реализационный период образца новой техники за счет ликвидации разрывов между отдельными фазами жизненного цикла, а также параллельного выполнения некоторых работ, относящихся к разным фазам.

Во-вторых, необходимо широкое использование технико-экономической оценки образцов новой техники по критерию "затраты — эффектив-

ность", что обеспечивает максимальную эффективность использования ресурсов, выделяемых на развитие науки и техники.

Итак, анализируя методологию проектирования образцов новой техники, включая программно-целевой метод, можно сделать ряд выводов:

1. Неопределенности в исходных данных — объективное свойство условий, в которых происходит процесс проектирования образцов новой техники. Причем эти неопределенности особенно велики на начальных фазах жизненного цикла (стадии предварительного проектирования); по мере перехода от одной фазы к другой уровень неопределенностей уменьшается.

2. Высокий уровень неопределенностей предполагает пофазное управление созданием образца новой техники, что, в свою очередь, требует количественной оценки ожидаемых характеристик образца, т.е. решения задач прогнозирования на каждой из рассматриваемых стадий проектирования (при различном уровне неопределенностей).

3. Качественное решение задачи проектирования возможно при использовании программно-целевого метода, основанного на широком использовании технико-экономической оценки образцов новой техники по различным критериям типа "затраты — эффективность".

Нетрудно заметить, что все сформулированные выводы содержат идею количественной оценки ожидаемых (прогнозируемых) характеристик объекта или образца новой техники. Причем задача прогнозирования должна решаться при крайне скудной информации о параметрах проектируемого образца новой техники. Чем точнее решается эта задача, тем можно точнее оценить ожидаемые параметры образца на различных фазах жизненного цикла, более качественно решить задачу проектирования объекта или образца новой техники (меньше стоимость разработки, короче реализационный период и т.д.). Таким образом, качество процесса проектирования образца новой техники и точность решения задачи прогноза параметров этого образца на различных стадиях проектирования оказываются тесно взаимосвязанными.

Прежде чем приступить к анализу и синтезу методов математического прогнозирования — а именно этим вопросам посвящена данная книга — коснемся хотя бы бегло эвристического прогнозирования, которое широко используется во всех сферах деятельности человека. Почему в век ЭВМ мы продолжаем говорить об эвристическом прогнозировании? В чем преимущество этого метода?

Метод эвристического (экспертного) прогнозирования оказывается достаточно надежным инструментом исследователя, конструктора, потому что он основан на способности человека принимать решение в обстановке жесточайшего дефицита информации. А именно в этой обстановке и проходит процесс проектирования образцов новой техники (особенно на стадии предварительного проектирования). Говоря о весьма важном начальном этапе прогнозирования, мы должны согласиться с К.Клаузевицем [16], который утверждал: "Умственная деятельность здесь покидает область строгого знания, логики и математики и превращается в искусство в более широком смысле этого слова, т.е. в умение

интуитивно выбирать из бесчисленного множества предметов и обстоятельств важнейшие и решающие. Большая или меньшая часть этой интуиции состоит в полусознательном сравнении всех величин и обстоятельств, с помощью которого быстро устраняется все маловажное и несущественное, а более нужное и главное распознается скорее, нежели путем строго логических умозаключений". Отсюда следует, что указанная способность человека является следствием функционирования его психики.

Одной из важнейших функций человеческой психики является моделирование той среды, в которой живет человек. Модель — это система, отражающая с упрощениями и неизбежными искажениями другую систему — объект или объекты, с которыми человек взаимодействует. Мышление в данном случае выступает как действие с моделями внутреннего и внешнего мира, направленное на достижение поставленных целей.

Для построения подобного рода моделей необходима информация, полученная либо в результате опыта, либо при обучении. Хранилищем этой информации является сознание, из которого человек извлекает нужные сведения, как бы "забывая" об огромном количестве иной информации, которая ему может понадобиться в другие моменты жизни, для построения других моделей внешнего мира. По поводу этой удивительной способности человеческого сознания академик А.Н. Колмогоров заметил [17], что "... вся сознательная жизнь человека устроена как-то своеобразно и сложно, но когда ее закономерности будут изучены, для моделирования ее потребуется гораздо меньше элементарных ячеек, чем для моделирования всего мозга".

Естественно предположить, что наш мозг строит адекватные внешнему миру модели, оперируя сравнительно небольшим количеством информации. Для этого на каждую предъявленную ситуацию в мозгу подготавливается несколько вариантов ответа и определенный механизм выбирает из них один наиболее целесообразный и адекватный данной ситуации. Еще до того как начинается действие, т.е. совершается тот или иной поступок, желаемое будущее уже существует в виде определенного результата этой деятельности — модели потребного будущего. Модель эта и служит той схемой, ориентируясь на которую проводится коррекция поведения, или при возникновении трудностей — коррекция целей. Именно эта особенность нашего мозга позволяет успешно работать в условиях неопределенности, в условиях огромного дефицита информации. Именно в силу такой особенности человек значительно превосходит существующие ЭВМ при решении подобного рода задач.

Однако следует признать, что возможности человека далеко не безграничны. При проектировании сложных систем, требующих обработки огромного числа альтернатив, только эвристические методы не обеспечивают в полной мере решение задачи выбора параметров образца новой техники.

Представим себе следующую ситуацию. Ответственным лицом (назовем его Главным конструктором) выдвинута идея, в целом определяющая образ будущего прибора. Начинается детальная конкретизация этой идеи, для чего мобилизуется совокупный профессиональный опыт разработчиков. Они перерабатывают массу документальной и справочной

информации, которую выдали им соответствующие информационные системы. При этом огромный труд затрачивается на поиск необходимых сведений, конкретных фактов в выданных статьях, патентах, отчетах. Документальная система не содержит в своей памяти организационных сведений, систематизированных под данную идею, под данный, еще не созданный образец новой техники. Эту систематизацию, мысленное конструирование (прогнозирование) будущего образца по аморфным сведениям, содержащимся в многочисленных документах, проделывает в своей памяти разработчик.

Приходится констатировать тот факт, что не существует банков данных по отдельным семействам техники, так как традиционно сложилась иная рубрикация документальных источников [14]. Например, известно, что число патентов, имеющих прямое или косвенное, но существенное отношение к летательным аппаратам, превысило 2 млн. [14]. Если собрать в одну библиотеку или банк данных всю информацию о летательных аппаратах, то возникают известные трудности, связанные с организацией "общения" с этим банком, даже при использовании мощных ЭВМ. Необходима сильная "компрессия" данных, и в этом направлении интенсивно ведутся научно-исследовательские работы. Так, в работе [14] предлагается формировать комбинаторные файлы, которые не хранят ни одного образца конструкции, а всякий раз конструируют нужный образ, определяя его конкретно-исторические или прогностические характеристики.

Из сказанного выше следует, что успешное решение задачи прогнозирования, а следовательно, и задачи проектирования образцов новой техники возможно при совместном использовании эвристических и математических методов, включая применение ЭВМ. Не рассматривая всю проблему комбинированного прогнозирования в целом, авторы акцентировали свое внимание только на ее части, а именно на вопросах математического прогнозирования (экстраполяции).

Поскольку речь идет о проектировании достаточно сложных систем, то экстремальные задачи системного проектирования приходится решать в многомерном пространстве. Существующие алгоритмы поиска оптимальных решений — детерминированные схемы градиента и наискорейшего спуска, алгоритмы линейного и нелинейного программирования, схемы случайного поиска и другие — при оптимизации сложных систем требуют построения моделей объекта и его среды, значительных затрат времени, большой загрузки вычислительных средств и групп оптимального проектирования. Между тем во многих конкретных технических и физических задачах проявляется сильная локальная зависимость математических решений от изменений исходных величин и проектных ситуаций. Близость решений имеет большое значение и может быть эффективно использована при решении задач прогнозирования (экстраполяции) характеристик разрабатываемого образца в условиях информационной недостаточности. Вопросам синтеза метода многомерной экстраполяции и применения такого метода для решения широкого круга задач проектирования и управления и посвящена эта книга.

Глава I

ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ПОСТРОЕНИЯ АЛГОРИТМА ПРИНЯТИЯ ОПЕРАТИВНЫХ РЕШЕНИЙ

1. МАТЕМАТИЧЕСКОЕ ПРОГНОЗИРОВАНИЕ

Математические методы прогнозирования широко используются в настоящее время. Их основными преимуществами являются объективность получаемой информации, высокая точность (при правильно выбранной модели), а также возможность автоматизации процесса прогнозирования с помощью современных вычислительных средств. Однако эти методы требуют наличия четко сформулированной математической модели поведения прогнозируемого объекта. Причем только в том случае, когда математическая модель правильно описывает поведение прогнозируемого объекта, мы вправе ожидать от математических методов точных результатов прогнозирования [41].

Таким образом, выбор и обоснование математической модели являются узловыми вопросами математического прогнозирования. Это в большинстве случаев далеко не тривиальная задача, и иногда она требует специальных исследований.

Выбор и обоснование модели прогнозируемого процесса представляют собой задачу идентификации, успех решения которой зависит от соотношения двух факторов — объема априорной информации о структуре и параметрах процесса (этот объем определяется степенью изученности данного явления, процесса) и объема измерительной (апостериорной) информации. Оба вида информации необходимы при синтезе модели, однако играют различные роли. Априорные сведения позволяют определить структуру модели, т.е. ее вид (число входов и выходов, характер связи между ними и т.д.). Эту процедуру обычно называют идентификацией в широком смысле или структурной идентификацией. Однако структура модели еще не сама модель, и для определения параметров модели необходимо выполнить измерения. Задачу определения этих параметров по наблюдениям за процессом при заданной структуре модели обычно называют идентификацией в узком смысле или параметрической идентификацией.

Качество решения задач идентификации и прогнозирования во многом определяется характером изучаемого процесса. И здесь вольно или невольно исследователь сталкивается с необходимостью классификации задач прогнозирования, поскольку различные процессы требуют разных методов исследования. Так, по характеру процесса различают прогнозирование детерминированных и случайных процессов.

Детерминированные процессы можно достаточно точно рассчитать заранее, поскольку случайные отклонения в них весьма малы. Примерами таких процессов могут служить движение небесных тел и простых механизмов (например, маятника), всякие изменения или перемещения, происходящие точно по расписанию или графику, и т.п.

При изучении детерминированных процессов возникают два типа задач, связанных с определением значений некоторой функции в интересующих нас точках по известным значениям этих функций в других точках.

Задача интерполирования заключается в нахождении значения функции внутри отрезка наблюдения. Сама функция может оставаться неизвестной, однако в большинстве случаев необходимо знать, к какому классу функций она относится: выражается ли прямой линией, параболой, гармонической функцией и т.п.

Задачей экстраполяции является нахождение значений функции в точке, лежащей вне отрезка наблюдения, по ее значениям внутри этого отрезка. Наиболее распространено линейное и параболическое экстраполирование, при этом функция выражается полиномом первой и второй степени.

Модели детерминированных процессов могут быть известны и выражены в виде аналитических функций, графиков или таблиц. Если мы имеем дело с вполне определенным физическим процессом (вид модели априорно известен), данные о котором не являются искаженными, задача прогнозирования может стать тривиальной. Например, не представляет труда решить задачу прогнозирования траектории движения шарика в заданном гравитационном поле при отсутствии силы сопротивления и известных координатах и начальной скорости.

Но чаще всего модели детерминированных процессов априорно неизвестны. В этом случае, если информация о процессе не искажена, задача прогнозирования может быть успешно решена, и все трудности, встречающиеся при ее решении, имеют лишь вычислительный характер. Задача значительно усложняется, если исследователь располагает весьма скудной информацией о наблюдаемом процессе. В реальной действительности встречается именно этот класс задач. Для подобного рода задач необходимо разрабатывать специальные методы, позволяющие решать либо задачу идентификации в условиях ограниченной информации, либо задачу прогнозирования по ограниченной информации без идентификации модели наблюдаемого процесса. Особый интерес эти методы представляют для решения задач прогнозирования многомерных процессов, поскольку многомерность является особенностью процессов и явлений окружающего мира.

В подавляющем большинстве будущие события, описываемые стохастическими моделями, можно предсказать на основе накопленного опыта лишь приближенно, в среднем. Точное предсказание невозможно по многим причинам и прежде всего в силу того, что исследователю могут быть известны не все факторы, участвующие в формировании анализируемого явления или процесса. Кроме того, неопределенность может оказаться и в основе самого физического явления, когда сложность и многочисленность причин не позволяют вычислить их суммарный эффект, когда не все условия задачи четко определены и т.д.

Однако каковы бы ни были причины случайности, при многократных наблюдениях исследуемого явления или процесса замечается так называемая статистическая устойчивость. Здесь вступает в силу закон больших

чисел, который, по определению академика А.Н. Колмогорова, представляет собой общий принцип, в силу которого совокупное действие большого числа случайных факторов приводит при некоторых весьма общих условиях к результату, почти не зависящему от случая [12]. Именно это положение лежит в основе теории статистических предсказаний.

Будем различать две задачи экстраполяции — пространственную и временную. Во втором случае задача экстраполяции сводится к прогнозированию поведения процесса во времени, т.е. по его наблюдаемому отрезку и на основе каких-то априорных данных следует оценить дальнейшее поведение процесса во времени. Этот вид задачи экстраполяции связан с прогнозированием и имеет в качестве аргумента один параметр — время. Успешность временного прогноза зависит в значительной мере от точности модели, описывающей прогнозируемый процесс. Временная экстраполяция в этой книге не рассматривается.

Пространственная экстраполяция связана с прогнозированием в пространстве. Это означает, что аргументом здесь является вектор, что, естественно, усложняет задачу прогноза. Этот вид экстраполяции состоит в оценке значений векторного поля по отдельным наблюдениям.

Содержательно задачу пространственной экстраполяции удобно описать в следующих терминах. Пусть имеется конечное множество ситуаций — точек в пространстве ситуаций, где определены некоторые решения (в общем виде — численные векторы). Задача экстраполяции состоит в том, чтобы оценить значение вектора решения в ситуации, которой не содержится в указанном множестве. Это нужно сделать, опираясь на относительное положение такой точки-ситуации среди точек, где известно решение.

Предложенный метод решения такой экстраполяционной задачи получил название метода многомерной линейной экстраполяции (МЛЭ). Но прежде чем рассмотреть МЛЭ, обратимся к двум наиболее распространенным задачам, требующим пространственной экстраполяции — проектированию и управлению, и проанализируем их связь с экстраполяцией такого рода.

2. ЭКСТРАПОЛЯЦИЯ В ПРОЕКТИРОВАНИИ И УПРАВЛЕНИИ

Задачу проектирования можно представить как задачу преобразования технического задания (обозначим его буквой X) в проект (обозначим его буквой Y). Процесс проектирования переводит X в Y , т.е. реализует преобразование:

$$Y = F^0(X), \quad (1)$$

где буквой F^0 обозначена процедура проектирования, т.е. получения проекта Y по его техническому заданию.

Для автоматизации процесса проектирования необходимо формализовать процедуру проектирования F^0 . Это можно сделать двояким образом.

С одной стороны, можно выявить правила проектирования, формализовать их и, запрограммировав соответствующим образом ЭВМ, получать с ее помощью проекты Y по техническим заданиям X . Однако, как легко заметить, этот подход требует детального изучения процесса проектирования и реализуется лишь для очень простых конкретных задач.

Можно поступить иначе. Обычно имеется опыт проектирования изделий такого рода, т.е. есть матрица из N прецедентов

$$I = \langle X_i, Y_i \ (i = 1, \dots, N) \rangle, \quad (2)$$

где X_i — i -е задание на проектирование, а Y_i — проект, разработанный "старым" способом по этому заданию, т.е.

$$Y_i = F^0(X_i) \ (i = 1, \dots, N).$$

Нельзя ли экстраполировать этот опыт на новое техническое задание X_{N+1} и получить хотя бы приблизительно, но сразу проект Y_{N+1} ? Оказывается можно! Разработанные методы экстраполяции позволяют формально решить эту сложную задачу без привлечения сведений о специфике процесса проектирования F^0 .

Задача управления объектом является также задачей преобразования. Исходным здесь будет описание состояния объекта управления и его среды. Обозначим эту информацию буквой X .

Информацию собирают с помощью системы датчиков, расположенных на объекте и в его среде. Решением, принимаемым на базе этой информации, является управление Y , которое следует реализовать в объекте, чтобы добиться заданных целей (они здесь предполагаются неизменными). Связь между X и Y в системе управления реализуется с помощью алгоритма управления F^0 , для работы которого необходимо располагать моделью объекта управления и т.д.

Этот алгоритм и позволяет определить в соответствии с зависимостью (1) управление Y по X .

Синтез такого алгоритма для сложных систем управления — чрезвычайно трудоемкая задача. Поэтому возникает вопрос: а нельзя ли здесь также воспользоваться таблицей прецедентов (2), где X_i — i -я ситуация, а Y_i — управление, с помощью которого, как показал опыт, была достигнута цель управления. То есть экстраполировать реализации зависимости (1) на новую ситуацию X_{N+1} с тем, чтобы получить управление Y_{N+1} , способное хотя бы приблизительно достигнуть целей управления, не используя модель объекта и т.д.

Как видно, обе задачи (проектирования и управления) близки и, как будет показано, могут быть решены экстраполяцией таблицы удачных прецедентов. Аналогично могут быть решены и другие задачи, опирающиеся на таблицу прецедентов. Среди них задача прогнозирования параметров динамической системы, распознавания образов, синтеза алгоритмов поиска корня и экстремума и многие другие.

Предлагаемый для этого метод МЛЭ универсален и прост при использовании для чрезвычайно широкого класса задач экстраполяции, далеко выходящих за рамки проектирования и управления.

3. ТРЕБОВАНИЯ. ПРЕДЪЯВЛЯЕМЫЕ К МЕТОДУ ЭКСТРАПОЛЯЦИИ

Итак, экстраполяция сводится к оценке значения \mathbf{y}_{N+1} по \mathbf{x}_{N+1} и матрице прецедентов (2)

$$\mathbf{y}_{N+1} = \Psi(\mathbf{x}_{N+1}, \mathbf{I}),$$

где Ψ – алгоритм экстраполяции.

Прежде чем определить требования к этому алгоритму, конкретизируем задачу и ограничимся следующим достаточно общим случаем: причина \mathbf{X} и следствие \mathbf{Y} измеряются в метрических шкалах и являются векторами

$$\mathbf{X} = (x_1, \dots, x_n),$$

$$\mathbf{Y} = (y_1, \dots, y_m),$$

определенными в соответствующих евклидовых пространствах размерности n и m , т.е.

$$\mathbf{X} \in R^n, \mathbf{Y} \in R^m.$$

В соответствии с этим таблица прецедентов (2) является числовой таблицей, которую удобно представить в виде числовой матрицы $N \times (n + m)$:

$$\mathbf{I} = \begin{Bmatrix} \mathbf{X}_1 & \mathbf{y}_1 \\ \dots & \dots \\ \mathbf{X}_N & \mathbf{y}_N \end{Bmatrix} = \begin{Bmatrix} x_{11} & x_{21} & \dots & x_{n1} & y_{11} & y_{21} & \dots & y_{m1} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ x_{1N} & x_{2N} & \dots & x_{nN} & y_{1N} & y_{2N} & \dots & y_{mN} \end{Bmatrix},$$

где

$$\mathbf{X}_i = (x_{1i}, \dots, x_{ni}),$$

$$\mathbf{y}_i = (y_{1i}, \dots, y_{mi}) \quad (i = 1, \dots, N).$$

Теперь сформулируем требования к методу экстраполяции.

1. При использовании метода должны точно восстанавливаться все следствия матрицы (протокола) \mathbf{I} , т.е.

$$\mathbf{y}_i = \Psi(\mathbf{X}_i, \mathbf{I}) \quad (i = 1, \dots, N).$$

Это значит, что при наличии ситуации \mathbf{X}_i , описанной в протоколе, должно быть точное значение следствия \mathbf{y}_i ($i = 1, \dots, N$).

2. Алгоритм Ψ экстраполяции должен быть векторным. Заметим, что в силу векторности алгоритма Ψ его можно представить в виде набора скалярных алгоритмов

$$\Psi = (\psi_1, \dots, \psi_m),$$

где ψ_j – алгоритм экстраполяции j -го следствия;

$$y_j = \psi_j(\mathbf{X}, \mathbf{I}) \quad (j = 1, \dots, m)$$

или

$$y_j = \psi(\mathbf{X}, \mathbf{I}_j) \quad (j = 1, \dots, m),$$

где ψ – единый скалярный алгоритм, а \mathbf{I}_j – фрагмент матрицы прецедентов;

$$I_j = \langle X_i, y_{ji} \rangle (i = 1, \dots, N) \rangle.$$

Такое представление позволяет свести векторный алгоритм Ψ к m -кратному повторению скалярного ψ . Это удобно при малых значениях m , но при больших m возникают вычислительные трудности. Требование 2 запрещает применение скалярных алгоритмов.

3. Трудоемкость реализации алгоритма должна расти по n и N не быстрее, чем линейно, а по m — медленнее, чем линейно. Это очень жесткое требование. Существующие методы экстраполяции, как правило, не удовлетворяют ему. Важность требования 3 очевидна, особенно при экстраполяции сложных явлений, где значения n и m велики.

4. Метод экстраполяции можно эффективно использовать и при малых объемах N таблицы прецедентов. Это означает, что алгоритм экстраполяции обязан работать с любым объемом N (вплоть до $N = 1$). Далеко не все существующие методы допускают сказанное.

5. Простота задания структуры причинно-следственной связи F . В ряде случаев, когда эта связь F проста, ее структуру задать легко, что однозначно определяет и структуру алгоритма экстраполяции. Но для сложных задач это сделать трудно. Поэтому при анализе метода экстраполяции важно учитывать сложность структуры связи F .

Как видно, не все требования одинаково важны для хорошего алгоритма экстраполяции. Так, например, требование 3 значительно важнее требования 2, а требования 1 и 4 иногда вообще можно снять. При "зашумленной" матрице теряет смысл требование 1, а при большом N нет необходимости в требовании 4.

Но при разработке универсальных алгоритмов экстраполяции с широкой областью применения (например, для пакета прикладных программ) данные требования необходимо удовлетворить, так как от этого зависит успех экстраполяции при решении конкретных задач.

Теперь рассмотрим существующие методы экстраполяции и выясним их соответствие указанным требованиям.

4. ЭКСТРАПОЛЯЦИЯ КАК СИНТЕЗ МОДЕЛИ (ОБЗОР ОСНОВНЫХ МЕТОДОВ)

Этот параграф посвящен анализу существующих методов экстраполяции и установлению их недостаточности для решения поставленной в п. 2 задачи.

Всякая экстраполяция (и интерполяция) опирается на некоторое формализованное представление о существующей связи F^0 (1) между причинами X и следствиями Y . (Здесь мы намеренно обобщаем схему, называя причиной измеряемые возмущения, а следствием — результат их действия.)

Формализация связи F^0 необходима прежде всего для однозначной оценки следствия Y при заданной причине X и для ее оптимизации по заданным критериям. Пусть φ — алгоритм такой формализации, включаю-

щий и оптимизацию. Тогда формализованную связь можно представить как результат переработки исходной информации алгоритмом φ

$$F = \varphi(I, J),$$

где I — протокол наблюдений различных реализаций формализуемой причинно-следственной связи F ; J — априорные представления о виде синтезируемой связи F (например, ее структура, порядок полинома и т.д.) и критериях эффективности F (например, минимум остаточной дисперсии и т.д.). Протокол I удобно записать как таблицу прецедентов (2). Он представляет собой записи реализаций причин и следствий в определенные дискретные моменты времени, связанных с F^0 -оператором реальной причинно-следственной связи (1).

Связь F^0 здесь всюду предполагается статической, что позволяет считать ее модель F функцией (точнее, векторной функцией векторного аргумента)

$$F = (f_1, \dots, f_m).$$

Это означает, что

$$y_i = f_i(x) \quad (i = 1, \dots, m),$$

т.е. задачу синтеза вектор-функции F всегда можно представить как m задач синтеза скалярных функций f_i .

Как видно, задача определения F по I и J есть задача выявления закономерности или задача синтеза модели [26]. Для решения этих задач разработаны многочисленные методы, которые могут быть использованы для задачи экстраполяции. Рассмотрим наиболее распространенные методы и проанализируем их специфику с позиций экстраполяции.

Различные методы синтеза F отличаются разными способами задания априорной информации.

1. Рассмотрим случай, когда J задает структуру F и критерий эффективности модели F . Пусть F_{st} — заданная структура формализуемой связи, т.е.

$$Y = F_{st}(X, C), \quad (3)$$

где $C = (c_1, \dots, c_k)$ — неизвестные параметры, которые определяются минимизацией заданного критерия эффективности

$$Q(F_{st}(X, C) | I) \rightarrow \min_{C \in S} \Rightarrow C^*.$$

Здесь S — область определения искомых параметров C (например, R^k — k -мерное декартово пространство), а C^* — решение этой задачи. Полученная таким образом модель $Y = F_{st}(X, C^*)$ и является формализацией искомой связи F^0 , которая используется при экстраполяции.

В качестве критерия эффективности Q обычно используют величину типа остаточной дисперсии, например,

$$Q = \sum_{i=1}^N |F_{st}(X_i, C) - y_i|^2.$$

Очевидно самым сложным здесь является задание структуры st связи F . Для того, как правило, необходимо располагать теорией этой причинно-следственной связи F^0 , что в значительной степени обесценивает

указанный подход. Действительно, в подавляющем большинстве практических задач относительно структуры оператора F сказать ничего нельзя, а поэтому и применять указанный подход нельзя. Требование 5 здесь не удовлетворяется.

Дело несколько облегчается в случае, когда оператор модели представляется в виде разложения по заданной системе функций.

II. Рассмотрим для простоты скалярный случай ($m = 1$), который тривиально обобщается на общий. Модель f при этом можно представить в виде

$$y = \sum_{j=1}^k c_j \varphi_j(X),$$

где

$$\Phi(X) = \{\varphi_1(X), \dots, \varphi_k(X)\} \quad (4)$$

— заданная система линейно-независимых функций, определяющая структуру модели.

Параметры C модели можно вычислить методом наименьших квадратов

$$Q(C) = \sum_{i=1}^N \left[\sum_{j=1}^k c_j \varphi_j(X_i) - y_i \right]^2 \rightarrow \min_c, \quad (5)$$

где используется протокол

$$I = \langle X_i, y_i (i = 1, \dots, N) \rangle.$$

Решение задачи (5), как известно, сводится к решению системы линейных алгебраических уравнений

$$\partial Q / \partial c_l = 0 \quad (l = 1, \dots, k), \quad (6)$$

коэффициенты которой определяют информационную матрицу

$$A = \left\| \sum_{i=1}^N \varphi_j(X_i) \varphi_l(X_i) \right\|_{k \times k}.$$

Очевидно, что для существования решения системы (6) необходимо, чтобы $N \geq k$. Оценим величину k для нелинейной системы функций (4), которая чаще всего используется в этом случае:

$$\varphi_l(X) = x_1^{\gamma_{1l}} x_2^{\gamma_{2l}} \dots x_n^{\gamma_{nl}} \quad (l = 1, \dots, k),$$

где γ_{il} ($i = 1, \dots, n$) — показатели степени — целые неотрицательные числа (включая ноль). Для линейной модели $\varphi_l(X) = x_{l-1}$ число неизвестных параметров C равно $k = n + 1$ ($l = 1, \dots, n + 1, x_0 = 1$). Для полной квад-

ратичной модели $k = 1 + n + \frac{n(n+1)}{2}$.

Как видно, с ростом размерности модели и ее порядка число k очень быстро растет. Так же быстро растет объем N необходимого протокола. Это обстоятельство сильно сужает применимость метода, так как нарушает требования 3 и 4, не говоря уже о том, что выбор системы представляет трудности, и серьезные (нарушение требования 5). Действительно, неудачный выбор системы функций (5) требует для хорошей модели

значительно увеличить число членов k , что в свою очередь заставляет увеличивать объем протокола I.

Для сложных объектов определить систему функций разложения априори обычно не удастся. Но существуют методы апостериорного определения полиномиальной структуры. К ним относится метод группового учета аргумента (МГУА) [13]. Рассмотрим его.

III. Синтез структуры модели F по МГУА происходит по этапам. На первом этапе образуются частные описания вида квадратичной формы

$$y^{(1i)} = \psi(x_l, x_j, C^{(1i)})$$

$$(i = 1, \dots, q > n), (l \neq j = 1, \dots, n), \quad (7)$$

где номера переменных l и j определяются либо полным перебором для малых n (и тогда $q = (n - 1)!$), либо для больших n случайным образом; $C^{(1i)}$ — параметры i -й частной модели на первом этапе, а ψ — простейшая квадратичная форма, например, вида

$$y^{(1i)} = c_1^{(1i)} + c_2^{(1i)} x_l + c_3^{(1i)} x_j + c_4^{(1i)} x_l x_j;$$

$C^{(1i)} = (c_1^{(1i)}, \dots, c_4^{(1i)})$ — параметры i -го частного описания определяются методом наименьших квадратов по протоколу (2). Далее следует селекция частных моделей.

На следующий этап отбираются n (или больше) тех частных моделей (7), которые имеют минимальную остаточную дисперсию.

На следующем втором этапе строятся аналогично q частных описаний вида

$$y^{(2i)} = \psi(y^{(1l)}, y^{(1j)}, C^{(2i)}), \quad (8)$$

где роль аргумента выполняют частные описания, прошедшие селекцию первого этапа. Как легко заметить, описания (8) являются полиномами четвертого порядка. Параметры $C^{(2i)}$ определяются методом наименьших квадратов по тому же протоколу (2). Далее следует селекция полученных частных описаний по критерию минимума дисперсии, и оставшиеся модели поступают на третий этап и т.д.

Легко заметить, что порядок моделей удваивается на каждом этапе. Эксперименты показывают, что МГУА дает возможность получить хорошие структуры моделей при малом объеме N протокола [13].

Однако МГУА является эвристическим методом, и, как во всяком эвристическом методе, его эффективность существенно зависит от удачного выбора ряда факторов (числа этапов селекции, дополнительных критериев, числа q частных моделей и т.д.).

Рассмотренные выше методы являются методами "сквозного" описания. Это означает, что строится одна формула модели, претендующая на описание поведения объекта во всей области S , охваченной протоколом I. Очевидно, что такой подход хорош лишь в том случае, когда объект ведет себя достаточно одинаково во всей области S . При наличии каких-то локальных отклонений сквозное описание плохо "справляется" со своей задачей. Нужно вводить локальные описания модели в районе той точки $(X_N + 1)$, где необходима экстраполяция.

IV. Локальный метод наименьших квадратов (ЛМНК) опирается на локальную модель в районе интересующей точки $Z = X_{N+1}$

$$Y = F_Z(X, C_Z), \quad (9)$$

которая отличается от (3) более простой структурой (знак структуры *st* здесь опущен). Очевидно, что параметры C_Z такой модели зависят от точки Z . Для выявления этой зависимости введем скалярную весовую функцию $\rho(W) \geq 0$ векторного аргумента W , максимум которой соответствует $W = 0$ и которая монотонно убывает с ростом $|W|$. Полагая $W = X - Z$, можно величиной $\rho(X - Z)$ определить меру важности информации протокола I в районе точки экстраполяции Z . Действительно, чем больше отличие X_i и Z , тем меньше информации содержит X_i для экстраполяции в Z .

Теперь параметры C_Z можно определить, минимизируя функцию невязки [15]:

$$\sum_{i=1}^N \rho(X_i - Z) [F(X_i, C_Z) - Y_i]^2 \rightarrow \min \Rightarrow C_Z^*. \quad (10)$$

Решение этой задачи C_Z^* и даст решение задачи экстраполяции

$$Y_{N+1} = F(X_{N+1}, C_{X_{N+1}}^*).$$

Ввиду локальности модели (9) ее структура может быть очень простой.

Приведем примеры двух весовых функций

$$\rho(W) = \frac{1}{1 + (1/\beta)|W|}, \quad \rho(W) = \exp[-W^2/\beta].$$

Легко заметить, что при $\beta \rightarrow \infty$ этот метод вырождается в метод наименьших квадратов, а при $\beta \rightarrow 0$ экстраполяция производится по ближайшей к X_{N+1} точке в протоколе I , т.е.

$$Y_{N+1} = Y_i, \text{ если } |X_i - X_{N+1}| = \min (i = 1, \dots, N).$$

Таким образом, варьируя параметр β , можно добиться разной степени локальности модели для экстраполяции.

Рассмотрим структуру модели в виде

$$F(X, C_Z) = \sum_{j=1}^k C_{jZ} \Phi_j(X).$$

И в самом простейшем случае локально-постоянной структуры

$$\Phi_1(X) = 1, k = 1$$

удается получить сразу после очевидных преобразований (дифференцирование функции (10) по неизвестным C_{1Z} и приравнивание результата нулю) выражение для модели

$$Y = C_{1Z} = \frac{\sum_{i=1}^N \rho(X_i - Z) Y_i}{\sum_{i=1}^N \rho(X_i - Z)}, \quad (11)$$

с помощью которой можно производить экстраполяцию в точке $Z = X_{N+1}$.

Легко заметить из (11), что в этом случае требование 1 выполняется лишь при $\beta \rightarrow 0$, т.е. при "сверхлокальной" экстраполяции (по ближайшему "соседу"). Заметим, что остальные требования, т.е. требования 2 – 5, здесь выполняются.

V. Использование сплайнов [38] дает возможность осуществлять кусочную аппроксимацию сложной функции. Обычно для этого применяют полиномы. Область S разбивается на несколько непересекающихся подобластей $S = \bigcup_{i=1}^q S_i$, в каждой из которых функция аппроксимируется полиномом степени не выше заданного числа p . На границах этих областей локальные аппроксимации стыкуют так, чтобы непрерывная производная была бы не ниже $p - \nu$ -го порядка, где $\nu > 0$ называют дефектом (обычно $\nu = 1$). Так, например, квадратичная сплайновая аппроксимация ($p = 2$) может быть реализована так, что на стыках непрерывна первая производная ($\nu = 1$) или непрерывна сама функция ($\nu = 2$). Необходимость локально-полиномиального описания при использовании сплайнов накладывает ограничения на размерность задачи экстраполяции, аналогичные тем, о которых говорилось в п. 3, хотя и снижает требования на порядок полинома.

Таким образом, использование сплайнов для экстраполяции удовлетворяет лишь требованиям 1 и 5.

В табл. 1 приведены данные по выполнению требований для рассмотренных методов. Плюс обозначает, что метод удовлетворяет соответствующему требованию, а минус – не удовлетворяет (отсутствие знака означает в зависимости от задачи и то, и другое). Здесь же приведены данные, полученные методом многомерной линейной экстраполяции, которому посвящена эта книга.

Т а б л и ц а 1

Методы	Требования				
	1	2	3	4	5
Оптимизация	+	–	–	–	–
МНК	+	–	–	–	–
МГУА	+	–	–	+	+
ЛМНК	–	+	+	+	+
Сплайны	+	–	–	–	+
МЛЭ	+	+	+	+	+

Из таблицы видно, что из известных методов наиболее подходят для решения задач экстраполяции ЛМНК и МГУА, что и оправдывается научно-производственной практикой их применения (особенно МГУА).

5. ЛИНЕЙНЫЕ ПРОСТРАНСТВА И ЛИНЕЙНЫЕ ПРЕОБРАЗОВАНИЯ

Для построения метода многомерной экстраполяции нам потребуются некоторые сведения из линейной алгебры, в частности, понятия линейного пространства, линейных преобразований и т.д.

Понятие линейного пространства, сложившееся в современной математике, обобщает ранее известные положения векторного анализа. С помощью векторов в физике изображаются силы, скорости, ускорения и т.п. Оттуда же взяты и правила действия с векторами. Так, например, под суммой двух векторов A и B понимают вектор, представляющий собой диагональ параллелограмма, построенного на A и B как на сторонах. При умножении вектора A на число λ получается вектор λA , длина которого равна длине A , умноженной на $|\lambda|$, а направление сохраняется, если $\lambda > 0$, и меняется на противоположное при $\lambda < 0$. Два вектора считаются равными, если параллельным переносом можно совместить их начала и концы. Кроме того, для удобства вводится так называемый нулевой вектор, т.е. такой, у которого конец совпадает с началом. Если к любому вектору прибавить нулевой вектор, то он не изменится, и если нулевой вектор умножить на произвольное число, то при этом получится снова нулевой вектор. Действия над векторами имеют ряд полезных легко проверяемых свойств. Например, $\lambda (A + B) = \lambda A + \lambda B$, $\lambda (\mu A) = (\lambda \mu) A$ и т.п.

Теперь, встав на аксиоматический путь, можно принять эти привычные свойства векторов на аксиомы. В результате получится абстрактное линейное (аффинное) или векторное пространство. Таким образом, линейное пространство — это множество объектов произвольной природы, которые мы называем векторами, удовлетворяющих некоторой довольно очевидной и естественной системе аксиом, основанной на рассмотренных ранее операциях сложения векторов и умножения вектора на число. И здесь, вполне понятно, может возникнуть вопрос: какой смысл имеют выражения $A + B$ и λA ? В случае обычных векторов на плоскости смысл этих символов ясен. Однако, как нам построить параллелограмм из элементов абстрактного множества? Что значит, например, удвоить длину абстрактного объекта? Единственно, что мы обязаны сделать — это указать правило, по которому любым двум элементам A и B из множества E ставится в соответствие третий элемент C , называемый их суммой и обозначаемый $A + B$. Тот факт, что элемент C выбирается из множества E , обозначается $C \in E$. Аналогично для любого числа λ и для любого элемента A мы должны указать элемент $B \in E$, который называется их произведением и обозначается λA . Этим, как говорят математики, мы на абстрактном множестве элементов $A, B \dots$ задаем операции сложения и умножения на число. Эти операции должны удовлетворять следующей естественной системе аксиом:

1. $A + B = B + A$ — коммутативность сложения;
2. $(A + B) + C = A + (B + C)$ — ассоциативность сложения;
3. $A + 0 = A$ — существование нуля;
4. $A + (-A) = 0$ — существование противоположного элемента;

5. $1 \cdot A = A$ — существование единицы (единичного элемента);
6. $(\lambda \mu) A = \lambda (\mu A)$ — ассоциативность умножения на число;
7. $\lambda (A + B) = \lambda A + \lambda B$ — дистрибутивность умножения числа на сумму элементов;
8. $(\lambda + \mu) A = \lambda A + \mu A$ — дистрибутивность умножения суммы чисел на элемент.

Здесь λ и μ — любые числа, символом 1 обозначен единичный элемент, а символом 0 — нулевой элемент.

Линейные пространства образуют обычные векторы на прямой, на плоскости и в обычном евклидовом пространстве. Это и неудивительно, поскольку именно по их образу и подобию мы строили абстрактное линейное пространство. Множество вещественных чисел с обычными операциями сложения и умножения также представляет собой вещественное линейное пространство. Роль нулевого элемента здесь играет обычный ноль.

Чтобы показать, что такое представление не является досужим мудрствованием, приведем пример линейного пространства, где элементами являются полиномы. Рассмотрим множество всех возможных полиномов степени не выше $n - 1$

$$A(x) = a_1 + a_2 x + \dots + a_n x^{n-1}$$

с вещественными коэффициентами a_1, \dots, a_n . Определим в этом множестве сложение и умножение на вещественное число обычным способом. Если $B(x) = b_1 + b_2 x + \dots + b_n x^{n-1}$, то $A(x) + B(x) = (a_1 + b_1) + (a_2 + b_2)x + \dots + (a_n + b_n)x^{n-1}$ и $\lambda A(x) = \lambda a_1 + \lambda a_2 x + \dots + \lambda a_n x^{n-1}$.

Легко убедиться, что эти операции удовлетворяют всем указанным выше аксиомам, где 0 и 1 — обычные ноль и единица. Следовательно, это множество образует вещественное линейное пространство [39].

Приведем еще пример — множество упорядоченных наборов из n вещественных чисел (a_1, a_2, \dots, a_n) . Введем операции сложения и умножения на число следующим образом: $(a_1, a_2, \dots, a_n) + (b_1, b_2, \dots, b_n) = (a_1 + b_1, a_2 + b_2, \dots, a_n + b_n)$; $\lambda (a_1, a_2, \dots, a_n) = (\lambda a_1, \lambda a_2, \dots, \lambda a_n)$.

Как легко убедиться, перед нами вновь линейное пространство с нулевым элементом вида $(0, 0, \dots, 0)$ и единичным $(1, 1, \dots, 1)$.

Свойства приведенных выше реализаций линейного пространства во многом аналогичны друг другу, поскольку в каждой из них выполняется одна и та же система аксиом. Однако между некоторыми реализациями есть одно существенное различие. Покажем это на примере обычных векторов на прямой и на плоскости. На первый взгляд кажется, что векторов на плоскости больше, чем на прямой. Но в буквальном смысле это неверно, поскольку и на прямой, и на плоскости векторов бесконечно много. Так в чем же дело? Оказывается линейное пространство обладает свойством, которое и выражает нашу интуитивную уверенность в том, что на плоскости "просторнее", чем на прямой. Покажем это.

Возьмем на прямой произвольный ненулевой вектор E . Тогда любой другой вектор A на этой прямой можно выразить в виде $A = \lambda E$, где λ — некоторое число (свое, разумеется, для каждого A). Если же мы возьмем ненулевой вектор E на плоскости, то найдутся векторы, которые нельзя записать подобным образом. Зато на плоскости есть, по крайней

мере, два ненулевых вектора E_1 и E_2 (на самом деле подобных пар бесконечно много) — таких, что любой другой вектор A запишется в виде

$$A = \lambda_1 E_1 + \lambda_2 E_2. \quad (12)$$

Если справедливо (12), то говорят, что вектор A равен линейной комбинации векторов E_1 и E_2 . В обычном евклидовом пространстве существует набор (система) из трех векторов E_1, E_2, E_3 — таких, что любой вектор A представляется в виде их линейной комбинации:

$$A = \lambda_1 E_1 + \lambda_2 E_2 + \lambda_3 E_3,$$

и нет системы с меньшим числом векторов, обладающей этим свойством. Аналогичную ситуацию в общем случае описывают, вводя понятие размерности линейного пространства.

По определению, линейное пространство E^n называется n -мерным, если в нем существует система из n векторов E_1, E_2, \dots, E_n , такая, что любой вектор A данного пространства представляется в виде

$$A = \lambda_1 E_1 + \lambda_2 E_2 + \dots + \lambda_n E_n = \sum_{i=1}^n \lambda_i E_i, \quad (13)$$

и нет системы из меньшего числа векторов, обладающей этим свойством. Система E_1, E_2, \dots, E_n называется базисом данного линейного пространства, а числа $\lambda_1, \lambda_2, \dots, \lambda_n$ — координатами вектора A в этом базисе. Тогда обычные векторы на прямой образуют одномерное линейное пространство, векторы на плоскости — двумерное и векторы в обычном пространстве — трехмерное линейное пространство.

Координаты λ_i в выражении (13) определяются однозначно — каждому вектору A соответствует один и только один набор координат $(\lambda_1, \lambda_2, \dots, \lambda_n)$ в данном базисе. В линейном пространстве E^n существует не один, а бесконечно много разных базисов. Это особенно наглядно видно на примере плоскости. Если вектор E_1 умножить на любое число $\alpha_1 \neq 0$, а E_2 на $\alpha_2 \neq 0$, то получится новый базис $\alpha_1 E_1, \alpha_2 E_2$. Если повернуть векторы E_1 и E_2 на произвольный угол, то снова получится некоторый базис. Да и вообще два ненулевых и непараллельных вектора на плоскости образуют базис.

С понятием базиса тесно связано понятие линейной независимости векторов. Конечная совокупность векторов E_1, E_2, \dots, E_n называется линейно независимой, если соотношение $\lambda_1 E_1 + \dots + \lambda_n E_n = 0$ имеет место только при $\lambda_1 = \lambda_2 = \dots = \lambda_n = 0$. В случае, когда можно найти такую совокупность скаляров $\lambda_1, \lambda_2, \dots, \lambda_n$, что хотя бы при одном из них, не равном нулю, справедливо соотношение $\lambda_1 E_1 + \lambda_2 E_2 + \dots + \lambda_n E_n = 0$, векторы E_1, E_2, \dots, E_n являются линейно зависимыми (любой E_i , для которого $\lambda_i \neq 0$, выражается через другие). Именно совокупность независимых векторов, через которую выражается любой вектор, называется базисом линейного пространства.

Таким образом, всякую систему из n векторов в пространстве E^n называют базисом в том случае, если выполняются следующие условия:

все векторы данной системы линейно независимы;

любой другой вектор пространства E^n , не входящий в эту систему, является линейной комбинацией векторов данной системы.

Вернемся к разобранным выше моделям линейного пространства. Пространство многочленов степени не выше $n - 1$ имеет размерность n . В качестве базиса можно взять, например, следующие векторы:

$$E_1 = x^0 \equiv 1, E_2 = x, E_3 = x^2, \dots, E_n = x^{n-1}. \quad (14)$$

Коэффициенты a_i многочлена $A(x) = a_1 + a_2 x + \dots + a_n x^{n-1}$ совпадают с координатами данного многочлена в базисе (14).

Пространство упорядоченных наборов из n чисел (a_1, a_2, \dots, a_n) также n -мерно. Векторы

$$\begin{aligned} E_1 &= (1, 0, \dots, 0); \\ E_2 &= (0, 1, \dots, 0); \\ E_n &= (0, 0, \dots, 1) \end{aligned} \quad (15)$$

образуют в этом пространстве базис.

Нашей конечной целью является описание процесса прогнозирования. А всякий прогноз опирается на модель объекта, т.е. на представления о связи между причинами и следствиями. В нашем случае это связь между ситуацией и решением, которое следует принимать в этой ситуации. Ситуации и решения образуют пространства, связь между которыми и следует выявить.

В дальнейшем понадобится два пространства: E^n и E^m размерностей n и m соответственно. Одно из пространств будет интерпретироваться как пространство ситуаций, состояний, наблюдений, а второе — как пространство решений, принимаемых в соответствующей ситуации или при данном наблюдении. Первое пространство характеризуется набором из n чисел, второе — набором их m чисел.

Предположим, что между этими пространствами E^n и E^m имеется некоторая связь или, говоря иначе, существует отображение одного пространства в другое (каждая ситуация отображается в решение, ей соответствующее). И здесь уместно рассмотреть, поскольку мы говорим о линейных пространствах, класс линейных отображений.

Пусть пространства E^n и E^m линейные. Тогда вполне уместен вопрос: аналогичны ли свойства этих пространств? Любые два линейных пространства изоморфны между собой [1, 39]. Другими словами, можно установить такое взаимное однозначное соответствие между векторами пространства E^n и E^m , при котором сумме векторов пространства E^n соответствует сумма их "образов" (отображений) в пространстве E^m , а произведению вектора на число в пространстве E^n соответствует произведение "образа" (отображения) этого вектора на то же самое число. Поясним это, используя рассмотренные ранее примеры.

Возьмем пространство многочленов $A(x)$ степени не выше $n - 1$, а также пространство упорядоченных наборов из чисел. Теперь сопоставим каждому многочлену $A(x) = a_1 + a_2 x + \dots + a_n x^{n-1}$ набор его коэффициентов (a_1, a_2, \dots, a_n) . Поскольку любой многочлен однозначно определяется набором своих коэффициентов, можно получить взаимно однозначное соответствие между двумя линейными пространствами

$$A(x) \leftrightarrow (a_1, a_2, \dots, a_n). \quad (16)$$

Пусть $A(x) = a_1 + \dots + a_n x^{n-1}$, а $B(x) = b_1 + \dots + b_n x^{n-1}$. Тогда $A(x) + B(x) = (a_1 + b_1) + (a_2 + b_2)x + \dots + (a_n + b_n)x^{n-1}$. В силу условия (16) многочлену $A(x) + B(x)$ соответствует набор $(a_1 + b_1, a_2 + b_2, \dots, a_n + b_n)$. Но, с другой стороны, $(a_1 + b_1, a_2 + b_2, \dots, a_n + b_n)$ является результатом применения операции сложения в пространстве наборов из n чисел. Следовательно, сумме многочленов соответствует сумма соответствующих числовых наборов. Аналогичные рассуждения могут быть приведены и относительно операции умножения на число, т.е. из выражения (16) следует

$$\lambda A(x) \leftrightarrow (\lambda a_1, \lambda a_2, \dots, \lambda a_n) = \lambda (a_1, a_2, \dots, a_n).$$

Таким образом, речь идет лишь о разных языках, на которых выражаются одинаковые факты. Любому утверждению в терминах числовых наборов точно соответствует аналогичное утверждение в терминах многочленов.

Запишем сказанное выше на языке математических символов, для чего введем следующие обозначения: $F: E^n \rightarrow E^m$, что означает отображение F пространства E^n в пространство E^m ; $Y = F(X)$ — образ $Y \in E^m$ элемента $X \in E^n$. Отображение $F: E^n \rightarrow E^m$ будет называться линейным, если оно имеет следующие свойства:

1. $F(X' + X'') = F(X') + F(X'')$ — аддитивность;
2. $F(\alpha X) = \alpha F(X)$ — однородность,

где $X, X', X'' \in E^n$, α — любое число.

Они эквивалентны одному такому свойству

$$F(\alpha X' + \beta X'') = \alpha F(X') + \beta F(X'').$$

Существуют общие методы построения линейных отображений, основанные на утверждении: каждому линейному отображению $F: E^n \rightarrow E^m$ однозначно ставится в соответствие такая матрица размера $m \times n$, что действие преобразования F на вектор $X \in E^n$ равносильно умножению данной матрицы на этот вектор. Справедливо и обратное утверждение: всякой матрице размера $m \times n$ однозначно соответствует линейное преобразование $F: E^n \rightarrow E^m$. Чтобы убедиться в справедливости этих утверждений, вспомним определение матриц и основные операции, которые с ними выполняются.

Прямоугольной матрицей размера $m \times n$ называются совокупность чисел, расположенных в виде прямоугольной таблицы, содержащей m строк и n столбцов:

$$\begin{vmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{vmatrix}. \quad (17)$$

Числа a_{ij} , составляющие данную матрицу, называются ее элементами; первый индекс элемента указывает номер строки, в которой расположен этот элемент, а второй индекс — номер столбца. Матрица, состоящая из одной строки или одного столбца, называется соответственно вектор-строкой или вектор-столбцом. Матрицы такого вида называют также просто векторами. Исходя из этого, будем обозначать матрицы пропис-

ными буквами латинского алфавита, т.е. сохраним обозначения, которые были приняты ранее для векторов.

Матрица, число строк и столбцов которой одинаково и равно n , называется квадратной матрицей порядка n . Совокупность $a_{ii} = (i = \overline{1, n})$ образует главную диагональ квадратной матрицы. Матрица, все элементы которой вне главной диагонали равны нулю, называется диагональной и кратко записывается в виде $A = \text{diag} \{a_1, a_2, \dots, a_n\}$. Если в диагональной матрице $a_1 = a_2 = \dots = a_n = 1$, то это единичная матрица n -го порядка, которая обозначается буквой I .

Определим основные операции над матрицами как элементами некоторого пространства.

Определение 1. Суммой двух прямоугольных матриц A и B одинаковых размеров называется матрица C тех же размеров, каждый элемент которой равен сумме соответствующих элементов матриц, т.е.

$$C = A + B, \quad (18)$$

если

$$C_{ij} = a_{ij} + b_{ij} \quad (i = \overline{1, \dots, m}; j = \overline{1, \dots, n}). \quad (19)$$

Из приведенного определения следует, что операция сложения матриц коммутативна, т.е. $A + B = B + A$, и ассоциативна, т.е. $(A + B) + C = A + (B + C)$. Она естественным образом распространяется на любое число слагаемых.

Определение 2. Произведение матрицы A на скаляр α есть матрица C , элементы которой получаются умножением соответствующих элементов матрицы A на число α , т.е. $C_{ij} = \alpha a_{ij}$. Очевидно, справедливы следующие соотношения:

$$\begin{aligned} \alpha (A + B) &= \alpha A + \alpha B; \quad (\alpha + \beta) A = \\ &= \alpha A + \beta A; \quad (\alpha \beta) A = \alpha (\beta A), \end{aligned}$$

где A и B — матрицы одинакового размера; α и β — числа (скаляры). Общий множитель элементов можно выносить за знак матрицы, считая его скалярным множителем.

Обе введенные операции не меняют размеров матриц, т.е. из исходных матриц размера $m \times n$ получали матрицы тех же размеров. Кроме того, эти операции превращают множества матриц в линейные пространства размера $m \times n$. Для доказательства сделанного вывода достаточно убедиться, что выполняются требования, которые накладываются на операции в линейном пространстве. Но над матрицами производятся и другие операции.

Определение 3. Преобразование матрицы A , состоящее в замене строк столбцами (или столбцов строками) при сохранении их нумерации, называется транспонированием. Полученная в результате такого преобразования матрица называется транспонированной и обозначается A^T . При транспонировании матрица размера $n \times m$ превращается в матрицу размера $m \times n$.

Существуют квадратные матрицы, которые совпадают со своими транспонированными, т.е. $A = A^T$. Такие матрицы называются симметрич-

ными и их элементы связаны очевидным соотношением $a_{ij} = a_{ji}$ (это симметрия относительно главной диагонали).

Операция транспонирования обладает следующими естественными свойствами:

- а) $(A^T)^T = A$ — двойное транспонирование не меняет матрицу;
- б) $(A^T + B^T) = (A + B)^T$ — операции транспонирования и суммирования можно менять местами;
- в) $(\lambda A)^T = \lambda A^T$ — скаляр не транспонируется.

Определение 4. Произведение матрицы A размера $m \times n$ (имеющей m строк и n столбцов) на матрицу B размера $n \times r$ (имеющей n строк и r столбцов) является матрица C размера $m \times r$, элемент c_{ij} которой равен сумме произведений элементов i -й строки матрицы A на соответствующие элементы j -го столбца матрицы B , т.е.

$$c_{ij} = \sum_{k=1}^n a_{ik} b_{kj} \quad (i = \overline{1, m}; j = \overline{1, r}). \quad (20)$$

Умножение A на B допустимо (произведение $A \cdot B$ существует), если число столбцов матрицы A равно числу строк матрицы B . В таких случаях говорят, что эти две матрицы согласуются по форме.

Для матриц A размера $m \times n$ и B размера $n \times m$ существует как произведение $A \cdot B$ размера $m \times m$, так и произведение $B \cdot A$ размера $n \times n$. Ясно, что при $m \neq n$ эти произведения не могут быть равными уже вследствие различных размеров результирующих матриц. Но даже при $m = n$, т.е. в случае квадратных матриц одинакового порядка, произведения $A \cdot B$ и $B \cdot A$ необязательно равны между собой. Отсюда следует, что вообще операция умножения матриц не подчиняется коммутативному закону, т.е. $A \cdot B \neq B \cdot A$. Если же выполняется соотношение $A \cdot B = B \cdot A$, то матрицы A и B называют коммутирующими или перестановочными.

Ассоциативный и дистрибутивный законы для матричного умножения выполняются во всех случаях, когда размеры матриц допускают соответствующие операции: $(A \cdot B) \cdot C = A \cdot (B \cdot C)$ (ассоциативность), $A \cdot (B + C) = A \cdot B + A \cdot C$ и $(A + B) \cdot C = A \cdot C + B \cdot C$ (дистрибутивность умножения слева и справа относительно сложения). Ассоциативность произведения позволяет не расставлять скобки в выражениях, содержащих произведение трех и более матриц, а дистрибутивность умножения дает возможность упростить формулы, содержащие матричные соотношения, облегчая тем самым вычисления. Нетрудно убедиться прямой подстановкой, что справедлива следующая формула, связывающая матричное произведение и операцию транспонирования:

$$A^T \cdot B^T = (B \cdot A)^T, \quad (21)$$

т.е. для вычисления произведения транспонированных матриц можно перемножить их в обратном порядке и транспонировать результат.

Умножение матрицы A размера $m \times n$ на единичную матрицу m -го порядка слева и на единичную матрицу n -го порядка справа не изменяет этой матрицы, т.е. $I_m A = A I_n = A$. Если хотя бы одна из матриц произведения $A \cdot B$ является нулевой, т.е. состоящей из одних нулей, то в

результате получим нулевую матрицу. Частный, но очень важный случай представляет собой умножение вектор-строки на вектор-столбец такой же длины. Нетрудно убедиться, что в результате такого умножения получается число (скаляр). Аналогичный результат получается при вычислении произведения $X^T \cdot Y$, где X^T и Y — вектор-столбцы одинаковой размерности. Такое умножение двух вектор-столбцов, так же как и результат их перемножения, носит название скалярного произведения двух векторов. Это произведение является важной характеристикой взаимного расположения векторов в пространстве.

Определение 5. Вектор-столбцы X и Y называются ортогональными (перпендикулярными), если $X^T Y = 0$, т.е. скалярное произведение равно нулю. При $n = 3$ это условие совпадает с условием ортогональности векторов в трехмерном пространстве, известном из аналитической геометрии. На основании этого определения нетрудно понять смысл выражения $X^T X$, называемого квадратом нормы (длины) вектора X :

$$X^T X = \|X\|^2 = x_1^2 + x_2^2 + \dots + x_n^2. \quad (22)$$

Из этого выражения при $n = 3$ можно найти длину трехмерного вектора.

Определение 6. Матрица A называется ортогональной, если

$$A^T A = I. \quad (23)$$

Ортогональная матрица размера 3×3 обладает следующим свойством: если в пространстве E^3 взять два ортогональных вектор-столбца и умножить их на эту матрицу, то в результате получим снова два ортогональных вектора. Это свойство легко распространяется на пространства произвольной размерности.

Вспомнив основные свойства матриц, можно вновь вернуться к ранее сделанному утверждению о взаимном соответствии линейного отображения и матрицы и доказать его математически.

Возьмем вектор $X \in E^n$ и умножим его слева на произвольную матрицу A размера $m \times n$. В результате получим вектор $Y \in E^m$, т.е. матрице A соответствует некоторое преобразование из E^n в E^m . Остается показать, что это преобразование линейное.

Поскольку умножение матриц дистрибутивно относительно их сложения, то можно записать

$$A(X + Y) = AX + AY.$$

Из определения умножения числа на матрицу следует, что

$$\alpha X = (\alpha x_1, \alpha x_2, \dots, \alpha x_n)^T.$$

Тогда, записывая произведение $A(\alpha X)$ по соответствующей формуле

$$y_i = \sum_{j=1}^n a_{ij} \alpha x_j = \alpha \sum_{j=1}^n a_{ij} x_j,$$

получим

$$Y = \alpha AX.$$

Таким образом, преобразование, соответствующее матрице A , есть линейное преобразование, поскольку оба условия, характеризующие

линейное пространство, выполнены. Это обстоятельство позволяет во всех вычислениях действие линейного преобразования на вектор заменить умножением этого вектора слева на вполне определенную матрицу (порядок умножения указан потому, что от него зависит результат). Говоря об умножении матриц, следует иметь в виду, что именно в этой операции заключен глубокий смысл преобразования (отображения) линейных пространств. Если первая матрица-сомножитель имеет четыре элемента в строке, а вторая матрица-сомножитель – два элемента, то строка матрицы, полученной в результате умножения, состоит также из двух элементов, т.е. в этом случае четырехмерное пространство отображается в двумерное. Благодаря правилу умножения матриц оказывается возможной матричная запись системы линейных уравнений, т.е. матричное равенство вида

$$\mathbf{y} = \mathbf{A}\mathbf{X} \quad (24)$$

является удобным представлением системы линейных уравнений:

[illegible]

Систему уравнений, записанную выше в матричной форме, можно рассматривать как линейное преобразование вектора X в вектор Y , которое определяется матрицей преобразования A . Решение матричного уравнения имеет вид

$$\mathbf{X} = \mathbf{A}^{-1} \mathbf{y}.$$

где A^{-1} – обратная матрица.

Обратная матрица, являясь аналогом обратного числа, имеет ряд особенностей. В обычной алгебре взаимно обратными называют два числа, произведение которых равно единице, т.е. $aa^{-1} = 1$. Аналогично в матричной алгебре две квадратные матрицы, произведение которых равно единичной матрице, т.е. $A \cdot A^{-1} = A^{-1} \cdot A = I$, называют взаимно обратными. Однако дальше этого аналогия не проходит.

Произведение $a^{-1} \cdot b$, где a и b — числа, можно представить как частное от деления b на a , но для матриц такое представление не имеет смысла, и в общем случае $A^{-1} \cdot B \neq B \cdot A^{-1}$. Поэтому вместо операции деления B на A вводят левое частное $A^{-1} \cdot B$ и правое частное $B \cdot A^{-1}$, которое сводится к умножению слева и справа на обратную матрицу A^{-1} .

Рассмотрим подробнее выражение \mathbf{AX} , где \mathbf{A} — матрица, \mathbf{X} — вектор. Матрицу \mathbf{A} можно представить состоящей из n вектор-столбцов

$$\mathbf{A} = (\mathbf{A}_1, \mathbf{A}_2, \dots, \mathbf{A}_n),$$

где $A_i = (a_{1i}, a_{2i}, \dots, a_{mi})^T, i = \overline{1, n}$.

Тогда

$$\mathbf{A} \mathbf{X} = (\mathbf{A}_1 x_1 + \mathbf{A}_2 x_2 + \dots + \mathbf{A}_n x_n),$$

т.е. $\mathbf{A}\mathbf{X}$ есть линейная комбинация векторов $\mathbf{A}_1, \mathbf{A}_2, \dots, \mathbf{A}_n$. Исходя из этого,

можно сформулировать задачу решения матричного уравнения следующим образом: требуется найти также коэффициенты линейной комбинации векторов $A_i, i = \overline{1, n}$, чтобы эта комбинация была равна заданному вектору Y . А как уже говорилось выше, с понятием линейной комбинации тесно связано важное понятие линейной независимости векторов.

Определение 7. Множество векторов A_1, A_2, \dots, A_n называется линейно независимым, если их линейная комбинация равна нулю тогда и только тогда, когда вектор, образованный из коэффициентов линейной комбинации, имеет только нулевые координаты (такой вектор называется нулевым). Интересно отметить, что максимальное число линейно независимых строк в произвольной матрице совпадает с максимальным числом линейных независимых столбцов. Это число называется рангом матрицы.

Итак, мы ввели целый ряд понятий и определений из линейной алгебры (понятие вектора, линейного пространства, матрицы, линейного преобразования, линейной комбинации векторов, ранга матрицы и т.д.), которые ниже будут использованы при описании метода многомерной экстраполяции.

6. АЛГОРИТМ МНОГОМЕРНОЙ ЛИНЕЙНОЙ ЭКСТРАПОЛЯЦИИ С ЕВКЛИДОВОЙ МЕТРИКОЙ

Алгоритм многомерной линейной экстраполяции опишем на примере решения задачи оптимального проектирования – типичной задачи, нуждающейся в экстраполяции.

Задача оптимального проектирования сложной системы заключается в определении параметров, доставляющих экстремум назначенному критерию качества системы в заданной ситуации. Под ситуацией можно понимать, например, исходные данные для проектирования: условия работы системы – ограничение на отдельные ее характеристики, технология производства и т.п.

Зависимость оптимальных параметров системы от ситуации, в которой проводится оптимизация, как правило, бывает неизвестной. Поэтому любые изменения в ситуации, типичные для процесса проектирования, приводят к необходимости проведения новых оптимальных расчетов для уточнения параметров системы. Такой способ корректировки оптимальных параметров значительно удлинняет процесс проектирования, поскольку каждый расчет, как правило, требует больших затрат времени.

Вполне естественно, возникает задача оценки оптимальных параметров по информации, которая накоплена в результате проектирования аналогичных систем или выполнения расчетов на оптимальность в ряде аналогичных ситуаций. При этом предполагается, что процесс нахождения параметров, близких к оптимальным в новой ситуации, должен происходить без проведения трудоемкой процедуры многопараметрической оптимизации.

Рассмотрим возможность решения этой задачи методом многомерной линейной экстраполяции [27, 28].

Пусть эффективность проекта определяется скалярной функцией двух векторных аргументов X и Y

$$Q = Q(X, Y), \quad (25)$$

представляющей собой критерий качества проектируемого объекта, где $Y = (y_1, \dots, y_m)$ – вектор оптимизируемых параметров объекта и $X = (x_1, \dots, x_n)$ – вектор ситуации, определяющий изменяющуюся в процессе проектирования специфику этого объекта и его среды, которые следует учитывать при оптимальном проектировании.

Для небольшого числа ситуаций X_i ($i = \overline{1, k}$) известны значения векторов оптимальных решений Y_i^* ($i = \overline{1, k}$), доставляющих экстремум критерию качества Q , т.е. предполагается решенной k раз оптимизационная задача

$$Q(X_i, Y) \rightarrow \text{extr} \Rightarrow Y_i^* (i = \overline{1, k}), \quad (26)$$

$$Y \in \Omega(X_i)$$

где $\Omega(X_i)$ – область изменения параметров Y объекта, зависящая в общем случае от ситуации X_i .

Ставится задача нахождения (без проведения трудоемкой оптимизации) вектора Y_{k+1} , близкого в определенном смысле к вектору Y_{k+1}^* , оптимальному в новой заданной ситуации X_{k+1} .

Пусть функция

$$Y^* = F(X). \quad (27)$$

обозначает неизвестную зависимость оптимальных параметров системы от ситуации. Тогда решение задачи (26) можно рассматривать как результаты k наблюдений этой неизвестной функции

$$X_i \rightarrow Y_i^*, i = \overline{1, k}. \quad (28)$$

где символ \rightarrow обозначает соответствие.

Следовательно, задача сводится к восстановлению неизвестной векторной функции F векторного аргумента X по конечному числу значений аргумента X и соответствующих ему значений вектора Y^* .

Множество векторов возможных ситуаций X обозначим через $\{X\}$, а соответствующее множество векторов решений Y – через $\{Y\}$.

Выбор способа решения поставленной задачи, очевидно, зависит от количества имеющейся информации о восстанавливаемой функции. Если число наблюдений k достаточно велико по сравнению с размерностью n вектора ситуации, задачу можно решить m -кратным применением различных методов аппроксимации: метода наименьших квадратов, метода потенциальных функций и т.д. Если число наблюдений k соизмеримо с размерностью вектора ситуации X и достаточно для построения базиса в пространстве $\{X\}$, можно построить линейную модель функции (27).

Рассмотрим случай, когда число k наблюдений мало и недостаточно для априорного построения линейной модели

$$Y^* = L(X), \quad (29)$$

т.е.

$$k < n + 1, \quad (30)$$

где n – размерность вектора X .

Для решения задачи в условиях информационной недостаточности, позволяющей из-за ограничений типа (30), поступим следующим образом:

построим линейную модель функции (29) на подпространствах $\{X'\}$ и $\{Y'\}$, образованных наблюдениями (28);

введем дополнительную гипотезу для доопределения функции на пространствах $\{X\}$ и $\{Y\}$.

Уравнения для элементов векторных подпространств (гиперплоскостей) $\{X'\}$ и $\{Y'\}$ имеют вид

$$\{X'\} = X_1 + \sum_{i=1}^{k-1} \lambda_i (X_{i+1} - X_1); \quad (31)$$

$$\{Y'\} = Y_1 + \sum_{i=1}^{k-1} \mu_i (Y_{i+1} - Y_1). \quad (32)$$

Предполагая решение Y оптимальным, будем использовать этот символ и далее вместо Y^* .

Линейную модель функции (27) на подпространствах $\{X'\}$ и $\{Y'\}$ строят, вводя следующую гипотезу.

Гипотеза 1. Преобразование $X' \rightarrow Y'$ линейно. Тогда для удовлетворения соответствия (28) следует положить

$$\lambda_i = \mu_i, \quad i = 1, \overline{k-1}. \quad (33)$$

Теперь задача восстановления функции (27) сводится к определению оператора преобразования $\{X\} \rightarrow \{Y\}$ на основе принятой гипотезы о линейности преобразования $\{X'\} \rightarrow \{Y'\}$. Если бы число наблюдений было больше $n+1$, то (учитывая, что отдельные наблюдения могут оказаться линейно зависимыми) при некотором k подпространство $\{X'\}$ совпало бы с линейной моделью пространства $\{X\}$, и задача восстановления функции (27) имела бы единственное решение — линейную модель функции. При наличии ограничения $k < n+1$ задача допускает бесчисленное множество решений. Эта неоднозначность решения устраняется введением дополнительной гипотезы.

Предварительно введем некоторую строго выпуклую скалярную функцию, которую назовем функцией близости

$$\Phi = \Phi(X, X'), \quad (34)$$

где $X \in \{X\}$, $X' \in \{X'\}$.

Каждой паре ситуаций X и X' эта функция ставит в соответствие число, характеризующее удаленность (близость) одной ситуации от другой.

Гипотеза 2. Каждому вектору $X \in \{X\}$ ставится в соответствие такой вектор $X' \in \{X'\}$, который доставляет минимум функции близости (34). Например, если функцию близости выбрать в виде

$$\Phi(X, X') = \|X - X'\|^2, \quad (35)$$

то гипотеза 2 приобретает следующий четкий геометрический смысл: каждому вектору ситуаций $X \in \{X\}$ ставится в соответствие его ортогональная проекция X' на линейное подпространство $\{X'\}$.

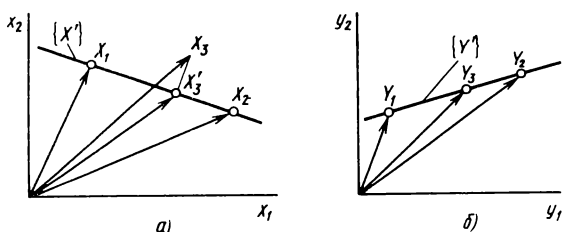


Рис. 1. Экстраполяция по двум обучающим векторам на плоскости:
а – пространство ситуаций; б – пространство решений

Параметры λ_i , определяющие вектор $X' \in \{X'\}$, ближайший в соответствии с функцией (34) к вектору $X \in \{X\}$, можно найти из системы уравнений

$$\partial \Phi / \partial \lambda_i = 0, i = \overline{1, k-1}. \quad (36)$$

Введение указанных гипотез позволяет построить следующий оператор преобразования $\{X\} \rightarrow \{Y\}$, т.е. алгоритм восстановления функции (27):

- 1) преобразуем $\{X\} \rightarrow \{X'\}$ с помощью гипотезы 2;
- 2) отображаем $\{X'\} \rightarrow \{Y'\}$ с помощью гипотезы 1;
- 3) отождествляем $\{Y'\} \equiv \{Y\}$.

Сокращенно алгоритм можно записать так:

$$\{X\} \rightarrow \{X'\} \rightarrow \{Y'\} \equiv \{Y\}.$$

Проиллюстрируем идею метода на простом примере двумерного пространства ситуаций.

Для построения алгоритма экстраполяции введем два линейных параметризованных пространства (рис. 1): двумерное пространство ситуаций X , где каждая ситуация определяется парой параметров x_1 и x_2 , и пространство решений Y , в котором каждое решение определяется двумя параметрами y_1 и y_2 . Далее предположим, что известны две проектные ситуации X_1 и X_2 , которым в пространстве решений соответствуют решения Y_1 и Y_2 . Задача ставится следующим образом: дана новая проектная ситуация X_3 , и нужно найти для нее решение Y_3 , располагая только указанной информацией. Решение этой задачи в соответствии с основной идеей метода многомерной экстраполяции будет иметь следующий вид.

Через известные ситуации X_1 и X_2 проведем прямую линию, являющуюся подпространством проектных ситуаций $\{X'\}$, а через точки Y_1 и Y_2 – линию, являющуюся подпространством решений $\{Y'\}$ (см. рис. 1). Для новой проектной ситуации X_3 определим ближайшую точку на подпространстве ситуаций, для чего опустим перпендикуляр из точки X_3 на линию $X_1 X_2$. Получим точку X_3' , являющуюся отображением точки X_3 на подпространстве ситуаций, которая делит отрезок $X_1 X_2$ в определенной пропорции. Разделив отрезок $Y_1 Y_2$ в пространстве $\{Y'\}$ в этой же пропорции, получим проектное решение Y_3 , соответствующее ситуации X_3 .

В соответствии с изложенным алгоритмом подпространства ситуаций и решений находим по следующим формулам:

$$\begin{aligned}\{X'\} &= X_1 + \lambda (X_2 - X_1), \\ \{Y'\} &= Y_1 + \mu (Y_2 - Y_1),\end{aligned}$$

где λ и μ — коэффициенты пропорциональности.

Гипотеза 1 о линейности преобразования $\{X'\} \rightarrow \{Y'\}$ в данном примере означает следующее: если ситуация X лежит на прямой $\{X'\}$, то соответствующий ей вектор оптимальных решений Y лежит на прямой $\{Y'\}$. Для удовлетворения исходным соответствиям

$$X_1 \rightarrow Y_1, X_2 \rightarrow Y_2$$

полагаем, что $\lambda = \mu$ (первое заданное соответствие удовлетворяется при $\lambda = \mu = 0$, второе — при $\lambda = \mu = 1$). Таким образом, введение гипотезы 1 в данном примере позволяет отыскивать вектор Y , оптимальный (с точностью до гипотезы линейности) в ситуации X , лежащей на прямой $\{X'\}$.

Если ситуация X не лежит на этой прямой (как, например, X_3), вводим меру близости между ситуациями (функцию близости). Ситуацию, не лежащую на прямой $\{X'\}$, отождествляем с ближайшей к ней ситуацией, лежащей на прямой. В этом заключается гипотеза 2. Если в данном примере функцию близости выбрать в виде (35), то ближайшей к ситуации X_3 окажется ситуация X_3' (проекция точки X_3 на прямую) доставляющая минимум функции близости. На основе гипотезы близости ситуацию X_3 отождествляем с ситуацией X_3' . Применяя гипотезу о линейности, находим вектор Y_3 , оптимальный в ситуации X_3' , по следующей приближенной формуле:

$$Y_3 = Y_1 + \mu (Y_2 - Y_1), \text{ где } \mu = \lambda.$$

Приведенная форма метода многомерной экстраполяции позволяет достаточно надежно решать задачу восстановления функции в условиях информационной недостаточности в пространствах малой размерности. С ростом размерности пространства растут вычислительные трудности, связанные с хранением больших массивов информации в памяти ЭВМ. Этих трудностей можно избежать, если вести вычисления рекуррентно [1, 2].

7. АНАЛИЗ ПОГРЕШНОСТЕЙ

МЕТОДА МНОГОМЕРНОЙ ЭКСТРАПОЛЯЦИИ ПРИ ВОССТАНОВЛЕНИИ ЛИНЕЙНЫХ И НЕЛИНЕЙНЫХ ОПЕРАТОРОВ СИСТЕМ

Необходимо подробнее остановиться на принципах линейности, компактности, близости, положенных в основу метода многомерной экстраполяции.

Применение принципа линейности означает, что мы предполагаем существование линейной зависимости выходных характеристик (эффективности, стоимости, массы и т.д.) сложной системы от управляемых технических параметров и проектных ситуаций. Как правило, эти зависимости в широком диапазоне изменения фазовых координат являются существенно нелинейными, но в ограниченной области проекти-

руемых параметров, проведя линеаризацию, можно выделить квазилинейную часть, и тогда указанный выше принцип линейности оказывается справедливым и полезным.

Принцип компактности предполагает, что новые проектные ситуации как-то примыкают к проектным ситуациям, ранее изученным, т.е. расположены от них "недалеко". Этот принцип позволяет отобразить новую ситуацию на известное подпространство ситуации с минимальной погрешностью, а затем также с минимальной погрешностью найти технический параметр, соответствующий новой ситуации, в условиях дефицита информации.

Из существа алгоритма следует, что погрешность метода экстраполяции оптимальных решений зависит от близости новых ситуаций к подпространству $\{X'\}$. Очевидно, что погрешность экстраполяции есть некоторая функция от нормы вектора близости и зависит от размерности проектных задач и полноты подпространства известных ситуаций.

Оценим аналитически ошибки экстраполяции на примерах восстановления методом МЛЭ линейных и нелинейных функций.

1. Ошибка восстановления линейной функции. Пусть зависимость вектора решения от вектора ситуации имеет линейный характер

$$y_i = \alpha_{i0} + \sum_{j=1}^n \alpha_{ij} x_j, \quad i = \overline{1, m}, \quad (37)$$

где α_{ij} — коэффициенты линейной формы.

Для простоты положим, что векторы ситуаций X_1, X_2, \dots, X_k , для которых известны векторы решений Y_1, Y_2, \dots, Y_k , единичны и ортогональны:

$$\begin{aligned} X_1 &= (1, 0, \dots, 0, \dots, 0) = e_1; \\ X_2 &= (0, 1, \dots, 0, \dots, 0) = e_2; \\ &\vdots \\ X_k &= (0, 0, \dots, 1, \dots, 0) = e_k. \end{aligned} \tag{38}$$

Оптимальные значения векторов $\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_k$ определим из (37):

$$\mathbf{y}_i = [(\alpha_{1_0} + \alpha_{1i}), (\alpha_{2_0} + \alpha_{2i}), \dots, (\alpha_{m_0} + \alpha_{mi})], i = \overline{1, k}. \quad (39)$$

Найдем рассмотренным выше методом выражение для вектора \mathbf{y} , близкого к оптимальному в произвольной ситуации:

$$\mathbf{X} = (x_1, x_2, \dots, x_k, \dots, x_n). \quad (40)$$

Для этого построим на векторах (38) линейное подпространство

$$\begin{aligned} \{\mathbf{X}'\} = & \mathbf{X}_1 + \sum_{i=1}^{k-1} \lambda_i (\mathbf{X}_{i+1} - \mathbf{X}_1) = (1 - \sum_{i=1}^{k-1} \lambda_i) \mathbf{e}_1 + \\ & + \sum_{i=1}^{k-1} \lambda_i \mathbf{e}_{i+1}. \end{aligned} \quad (41)$$

Функция близости (35) принимает вид

$$\Phi(\mathbf{X}, \mathbf{X}') = (x_{i-1} + \sum_{j=1}^{k-1} \lambda_j)^2 + \sum_{i=1}^{k-1} (x_{i+1} - \lambda_i)^2 +$$

$$+ \sum_{j=k+1}^n x_j^2. \quad (42)$$

Коэффициенты λ_i вектора \mathbf{X}' , ближайшего в соответствии с функцией (42), найдем из системы уравнений (36), которая для выбранной функции близости оказывается линейной:

$$\partial\Phi/\partial\lambda_i = (x_1 - 1 + \sum_{j=1}^{k-1} \lambda_j) - (x_{i+1} - \lambda_i) = 0, i = \overline{1, k-1}. \quad (43)$$

Преобразуя систему (43), получим

[illegible]

Система (44) имеет единственное решение, так как ее определитель Δ не равен нулю. Действительно, после несложных преобразований окончательно получим

$$\Delta = k. \quad (45)$$

Для нахождения решения системы (44) сделаем в ней следующие тождественные преобразования; вычтем из первого уравнения второе, третье и т.д.

$$\begin{aligned}\lambda_2 &= \lambda_1 - x_2 + x_3; \\ \lambda_3 &= \lambda_1 - x_2 + x_4; \\ &\dots\dots\dots \\ \lambda_{k-1} &= \lambda_1 - x_2 + x_k.\end{aligned}\tag{46}$$

Подставляя (46) в последнее уравнение системы (44), получим окончательно

$$\lambda_i = (1 - \sum_{j=1}^k x_j)/k + x_{i+1}, i = \overline{1, k-1}. \quad (47)$$

Вектор X' , ближайший в соответствии с функцией (42) к вектору X , получим, подставляя (47) в (41):

$$\begin{aligned} \mathbf{X}' = & (x_1 + \frac{1 - \sum_{j=1}^k x_j}{k}) e_1 + (x_2 + \frac{1 - \sum_{j=1}^k x_j}{k}) e_2 + \dots \\ & \dots + (x_k + \frac{1 - \sum_{j=1}^k x_j}{k}) e_k. \end{aligned} \quad (48)$$

Вектор Y с точностью до принятых гипотез, оптимальный в ситуации X , получим, подставляя (47) в (32):

$$y_i = (\alpha_{i0} + \sum_{j=1}^k \alpha_{ij} x_j) + \frac{(1 - \sum_{j=1}^k x_j)}{k} \sum_{i=1}^k x_{ij}, i = \overline{1, m}. \quad (49)$$

Как видно, выражения (37) и (49) различны, т.е. восстановление функции проведено с ошибкой.

Ограничим область, в которой проводится восстановление функции, n -мерным единичным кубом

$$0 \leq x_j \leq 1, j = \overline{1, n} \quad (50)$$

и найдем в этой области среднеинтегральную и максимальную локальную ошибки восстановления функции (37).

Локальной ошибкой назовем величину

$$\delta = |\mathbf{y}^* - \mathbf{y}|^2, \quad (51)$$

где \mathbf{y}^* — точное значение вектора оптимальных параметров, определяемое из (37); \mathbf{y} — экстраполированное значение вектора параметров системы.

Подставляя (37) и (49) в (51), находим выражение для локальной ошибки

$$\delta = \sum_{i=1}^m \left[\sum_{j=k+1}^n \alpha_{ij} x_j + \frac{\left(\sum_{j=1}^k x_j - 1 \right)}{k} \sum_{j=1}^k \alpha_{ij} \right]^2. \quad (52)$$

В области (50) это выражение достигает максимума при $x_j = 1, j = \overline{1, n}$

Таким образом, максимальная локальная ошибка восстановления функции (37) в области (50) соответствует ситуации $\mathbf{A}_n (1, 1, \dots, 1)$ и равна

$$\delta_{\max} = \sum_{i=1}^m \left[\sum_{j=k+1}^n \alpha_{ij} + (1 - 1/k) \sum_{j=1}^k \alpha_{ij} \right]^2. \quad (53)$$

Интегральную ошибку метода определим как среднюю локальную в области (50)

$$\delta_F = \int_{\{S\}} |\mathbf{y}^* - \mathbf{y}|^2 dS. \quad (54)$$

Подставляя (52) в (54) и учитывая (50), получим

$$\delta_F = \int_0^1 dx_1 \int_0^1 dx_2 \dots \int_0^1 \left\{ \sum_{i=1}^m \left[\sum_{j=k+1}^n \alpha_{ij} x_j + \frac{\left(\sum_{j=1}^k x_j - 1 \right)}{k} \sum_{j=1}^k \alpha_{ij} \right]^2 \right\} dx_n.$$

После вычисления интегралов выражение (54) принимает вид

$$\delta_F = \sum_{i=1}^m \left[\frac{1}{4} \left(\sum_{j=k+1}^n \alpha_{ij} \right)^2 + \frac{1}{12} \sum_{j=k+1}^n (\alpha_{ij})^2 + \right. \\ \left. + \frac{(k-2)}{2k} \sum_{j=1}^k \alpha_{ij} \left(\sum_{j=k+1}^n \alpha_{ij} \right) + \frac{3k^2 - 11k + 12}{12k^2} \left(\sum_{j=1}^k \alpha_{ij} \right)^2 \right]. \quad (55)$$

Проанализируем полученные результаты. Пусть для простоты в выражении (37)

$$\alpha_{ij} = \begin{cases} 1 & \text{при } i = j, \\ 0 & \text{при } i \neq j, \end{cases}$$

т.е. векторы $\mathbf{Y}_1^*, \mathbf{Y}_2^*, \dots, \mathbf{Y}_k^*$, как и векторы ситуации (38), единичны и ортогональны. Тогда выражения для δ_{\max} и δ_F принимают следующий вид:

$$\delta_{\max} = \frac{(k-1)^2}{k} + (n-k); \quad (56)$$

$$\delta_F = \frac{1}{3} (n-k) + \frac{3k^2 - 11k + 12}{12k}. \quad (57)$$

Приведем (57) к виду

$$\delta_F = \frac{1}{3} n + \frac{-k^2 - 11k + 12}{12k}. \quad (57')$$

Второе слагаемое в этом выражении увеличивается по модулю с ростом k , оставаясь при любом $k > 1$ отрицательным. Следовательно, при увеличении числа k наблюдений от двух (минимальное число наблюдений) до n ошибка δ_F монотонно уменьшается.

Для сравнения рассмотрим простейший метод восстановления функции, при котором новая ситуация отождествляется с ближайшей к ней известной ситуацией (назовем его методом "ближайшего соседа"). Максимальная локальная ошибка такого метода восстановления функции в области (50), очевидно,

$$\delta_{\max}^* = (k-1) + (n-k) = n-1. \quad (58)$$

Сравнивая (56) и (58), видим, что по этому критерию МЛЭ тем лучше, чем больше k .

Среднеинтегральная ошибка метода "ближайшего соседа" определяется выражением

$$\delta_F^* = k \int_0^1 dx_1 \int_0^1 dx_2 \dots \int_0^1 dx_k \int_0^1 dx_{k+1} \dots \\ \times \int_0^1 [(x_1-1)^2 + x_2^2 + \dots + x_n^2] dx_n,$$

или после вычисления интегралов

$$\delta_F^* = \frac{1}{3} (n-k) + \frac{k^3 - k + 6}{3(k^2 + 3k + 2)}. \quad (59)$$

Для сравнения δ_F и δ_F^* вычтем (57) из (59):

$$\delta_F^* - \delta_F = \frac{k^4 + 2k^3 + 11k^2 + 10k - 24}{12k(k+1)(k+2)}. \quad (60)$$

Как видно, при любом $k > 1$ выражение $(\delta_F^* - \delta_F) > 0$. Следовательно, среднеинтегральная ошибка рассмотренного выше метода восстанов-

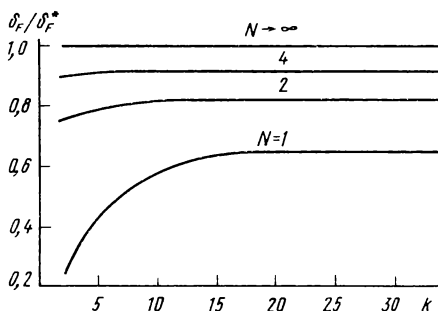
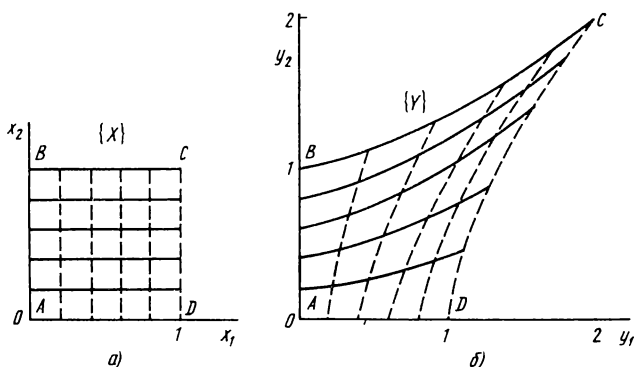


Рис. 2. Зависимость ошибки восстановления линейной функции от числа наблюдений

Рис. 3. Пример нелинейной зависимости $Y = Y(X)$:

a — ортогональная сетка в пространстве ситуаций; b — криволинейная сетка в пространстве управляемых параметров



ления функции монотонно меньше ошибки "ближайшего соседа". Как видно из графика зависимости отношения среднеинтегральных ошибок методов от числа наблюдений k при различном соотношении $N = n/k$ размерности пространства ситуаций n и числа наблюдений k (рис. 2), оба способа восстановления линейной функции равноценны в случае, когда $k \ll n$ ($N \rightarrow \infty$). Этот факт можно объяснить так: если исходная информация ничтожно мала, то любая разумная процедура восстановления функции на основе наблюдений даст одинаково плохой результат.

2. **Ошибка восстановления нелинейной функции.** Пусть зависимость вектора решений от вектора ситуаций имеет нелинейный вид

$$y_i^* = x_i (1 - x_i^2 + \sum_{j=1}^n x_j^2), \quad i = \overline{1, m}. \quad (61)$$

Для двумерного случая (рис. 3) преобразование (61) переводит ортогональную сетку в пространстве $\{X\}$ в криволинейную сетку в пространстве $\{Y\}$.

Векторы ситуаций, для которых известны оптимальные значения векторов управляемых параметров $Y_1^*, Y_2^*, \dots, Y_k^*$, пусть, как и в предыдущем примере, единичны и ортогональны. Векторы $Y_1^*, Y_2^*, \dots, Y_k^*$ определяются из зависимости (61). В данном примере они, как видно, тоже будут единичны и ортогональны.

Найдем выражение для вектора X , близкого к оптимальному в новой ситуации (40). Линеаризуя подпространство $\{Y'\}$ и применяя полученные в линейном примере результаты, получим

$$y_i = x_i + \frac{1 - \sum_{j=1}^k x_j}{k}, \quad i = \overline{1, m}. \quad (62)$$

Найдем среднеинтегральную ошибку восстановления функции (61) в области (50)

$$\delta_F = \int_0^1 dx_1 \int_0^1 dx_2 \dots \int_0^1 \left\{ \sum_{i=1}^m [a_i (1 - a_i^2 + \sum_{j=1}^n a_j^2) - (a_i + \frac{1 - \sum_{j=1}^n a_j}{k})]^2 \right\} da_m.$$

и после вычисления интегралов получим

$$\delta_F = \frac{(n-k)}{3} \left[\frac{(n+4)(5n+4)}{45} \right] + \frac{3k^2 - 11k + 12}{12k} + \frac{(20n^2 + 66n - 41)k - 150n + 105}{540} \quad (63)$$

Приведем (63) к виду

$$\delta_F = \frac{20n^3 + 96n^2 - 86n + 105}{540} - \frac{30k^2(n-1) + 495(k-1,09)}{540k}.$$

Второе слагаемое в этом выражении увеличивается по модулю с ростом k (при фиксированном n), оставаясь при любом $k > 1$ отрицательным. Следовательно, и в нелинейном случае справедливо утверждение, что при увеличении числа наблюдений от двух до n ошибка δ_F восстановления функции рассмотренным методом монотонно уменьшается.

Для сравнения найдем ошибку δ_F^* восстановления функции (61) методом "ближайшего соседа":

$$\delta_F^* = k \int_0^1 dx_1 \int_0^1 dx_2 \dots \int_0^1 dx_k \int_0^1 dx_{k+1} \dots \times \\ \times \int_0^1 \left\{ [x_1 (1 - x_1^2 + \sum_{j=1}^n a_j^2) - 1]^2 + \sum_{i=2}^m [a_i (1 - a_i^2 + \sum_{j=1}^n a_j^2)]^2 \right\} dx_m.$$

или после вычисления интегралов

$$\delta_F^* = \frac{n}{3} + \frac{k(k-1)(3n-2k+4)}{27} + \frac{k(k-1)(n+4)}{15(k+4)} - \\ - \frac{(k-1)(2k^2-5k+9)}{3(k+1)(k+3)} + \frac{(n-k)}{3} \left[\frac{k(23k+8)}{15(k+1)} + \right. \\ \left. + \frac{(n-k-1)(5n+10k+29)}{45} \right]. \quad (64)$$

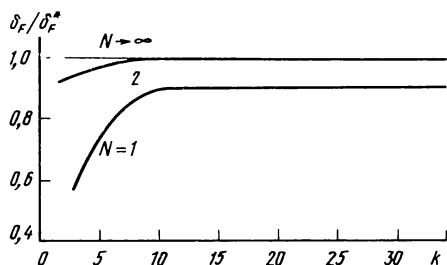


Рис. 4. Зависимость ошибки восстановления нелинейной функции от числа наблюдений

Вычитая (63) из (64), находим, что $(\delta_F^* - \delta_F) > 0$ при любом $n \geq k > 1$. Следовательно, и в нелинейном случае среднеинтегральная ошибка рас-

смотренного метода восстановления функции монотонно меньше ошибки "ближайшего соседа". На рис. 4, где приведена зависимость отношения ошибок δ_F/δ_F^* от числа наблюдений k при различном соотношении $N = n/k$, видно, что оба способа восстановления нелинейной функции равноценны в случае, когда $k \ll n$. Однако отношение δ_F/δ_F^* здесь быстрее стремится к единице при увеличении n по сравнению с k , чем в предыдущем примере. Это объясняется тем, что на ошибку восстановления функции по недостаточному числу наблюдений в этом случае накладывается дополнительная ошибка линеаризации нелинейного пространства.

В заключение отметим, что рассмотренный метод применим для восстановления линейной функции в случае малого числа наблюдений, недостаточного для построения базиса в пространстве аргументов. Если восстанавливаемая функция нелинейна, метод можно применять при любом числе наблюдений. В этом случае значение функции в новой ситуации определяется не по всем имеющимся наблюдениям, а лишь по ближайшим к новой ситуации. В результате осуществляется локально-линейное приближение нелинейной функции.

8. АЛГОРИТМ МНОГОМЕРНОЙ ЛИНЕЙНОЙ ЭКСТРАПОЛЯЦИИ НА ПОДВИЖНЫХ ПОДПРОСТРАНСТВАХ СИТУАЦИЙ И РЕШЕНИЙ

В основу исходного алгоритма многомерной линейной экстраполяции положено предположение о том, что образующее подпространство проектных ситуаций $\{X\}$ фиксированно и неподвижно. В этом случае алгоритм можно трактовать так: имеются неподвижные и фиксированные векторы X_1 и X_2 , характеризующие пространство ситуаций; необходимо найти оптимальный множитель λ , минимизирующий расстояние в евклидовой метрике между новой проектной ситуацией X_3 и известным подпространством $\{X\}$. Как показывают приведенные выше примеры, алгоритм многомерной экстраполяции, как правило, обеспечивает достаточно высокую точность решения задач, связанных с принятием оперативных решений в условиях неполной информации. Этот вывод тем более справедлив, чем полнее выполняются требования, вытекающие из существования метода (речь идет о выполнении гипотезы линейности пространства, выборе обучающих последовательностей и т.д.). Если же изложенные

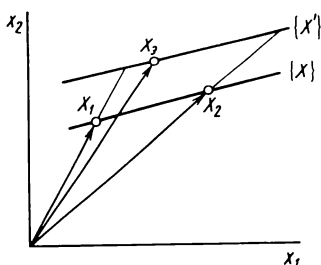


Рис. 5. Метод сдвига подпространства ситуаций

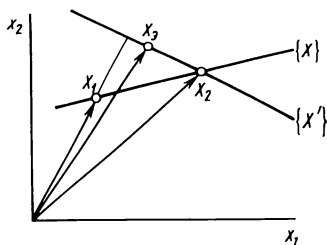


Рис. 6. Метод поворота подпространства ситуаций

выше требования выполнены не полностью и при постановке задачи, которая должна решаться методом многомерной экстраполяции, допущен некоторый "произвол", в частности, "неудачно" выбраны обучающиеся последовательности, то решение может быть получено с большой ошибкой. И здесь вполне уместно рассмотреть пути построения модифицированных алгоритмов, обеспечивающих более высокую точность решения экстраполяционных задач.

Для повышения точности линейной экстраполяции, применяя аксиомы линейного пространства, можно несколько иначе подойти к формированию подпространства проектных ситуаций.

В частности, используем идею подвижных векторных оболочек и попытаемся сдвинуть или повернуть подпространство ситуаций таким образом, чтобы сблизить подпространство $\{X'\}$ с новым вектором X_3 (рис. 5, 6). Рассмотрим подробнее оба метода [9].

Сдвигая подпространство ситуаций, введем параметр p и образуем векторы pX_1 и pX_2 . Тогда подпространство ситуаций можно записать в виде

$$\{X'\} = pX_1 + \lambda(pX_2 - pX_1) = p[X_1 + \lambda(X_2 - X_1)]. \quad (65)$$

Далее построим функцию близости $\Phi(X' - X_3)$, которая будет зависеть от параметров p и λ :

$$\Phi(X' - X_3) = \{p[X_1 + \lambda(X_2 - X_1)] - X_3\}^2. \quad (66)$$

Для минимизации этой строго выпуклой функции достаточно найти производные по параметрам p и λ и приравнять их нулю, т.е.

$$\Phi_{\lambda}' = 2\{p[X_1 + \lambda(X_2 - X_1)] - X_3\}[p(X_2 - X_1)] = 0; \quad (67)$$

$$\Phi_p' = 2\{p[X_1 + \lambda(X_2 - X_1)] - X_3\}[X_1 + \lambda(X_2 - X_1)] = 0. \quad (68)$$

Из рис. 5 следует, что в двумерном пространстве при λ и p , найденных в результате решения уравнений (67) и (68), значение близости станет не только минимальным, но будет равным нулю, т.е. задача экстраполяции будет решена без погрешности. Естественно, что при обучающей последовательности из двух векторов в трехмерном пространстве и в пространстве большей размерности это уже не будет справедливо, т.е. в общем случае будет иметь место некоторая конечная погрешность решения задачи экстраполяции.

Полученные формулы легко обобщаются на многомерный случай, т.е. на случай, когда известно несколько проектных ситуаций X_i , $i = 1, k$:

$$\{X'\} = p [X_1 + \sum_{i=1}^{k-1} \lambda_i (X_{i+1} - X_1)]; \quad (69)$$

$$\Phi(X' - X_3) = \sum_{l=1}^n \left\{ p [X_{1l} + \sum_{i=1}^{k-1} \lambda_i (X_{i+1l} - X_{1l}) - X_{3l}] \right\}^2. \quad (70)$$

Аналогично двумерному случаю решаем и задачу минимизации функции (70) и находим значения p и λ_i .

Экстраполированное решение получаем по формуле

$$Y_3 = p [Y_1 + \sum_{i=1}^{k-1} \lambda_i (Y_{i+1} - Y_1)]. \quad (71)$$

Возможна и другая форма записи уравнения (65), легко получаемая после несложных преобразований:

$$\{X'\} = pX_1 + \lambda (pX_2 - pX_1) = X_1 (p - \lambda p) + X_2 \lambda p.$$

Обозначая $p - \lambda p = \lambda_1^*$, $\lambda p = \lambda_2^*$, получим уравнение для подпространства ситуаций следующего вида:

$$\{X'\} = \lambda_1^* X_1 + \lambda_2^* X_2 = \sum_{i=1}^2 \lambda_i^* X_i. \quad (72)$$

В многомерном случае, т.е. при наличии в обучающей последовательности k векторов, формула (72) будет иметь вид

$$\{X'\} = \sum_{i=1}^k \lambda_i^* X_i. \quad (73)$$

Этому случаю соответствует функция близости

$$\Phi(X' - X_3) = \sum_{l=1}^n \left[\sum_{i=1}^k \lambda_i^* X_{il} - X_{3l} \right]^2. \quad (74)$$

Далее техника решения экстраполяционной задачи остается прежней, т.е. минимизируем квадратичную форму (74), решаем систему уравнений, в результате чего находим параметры λ_i^* .

Аналогичного результата можно достичь поворотом подпространства проектных ситуаций (см. рис. 6). Осуществляя поворот подпространства ситуаций, введем параметр p для ситуации X_1 . Тогда запись подпространства ситуаций примет вид

$$\{X'\} = pX_1 + \lambda (X_2 - pX_1). \quad (75)$$

Функцию близости запишем в следующей форме:

$$\Phi(X' - X_3) = [pX_1 + \lambda (X_2 - pX_1) - X_3]^2. \quad (76)$$

Минимизируя функцию близости, найдем параметры p и λ . Из рис. 6 следует, что при некоторых p и λ можно свести функцию близости, а следовательно, и погрешность экстраполяции к нулю.

В многомерном случае формулы (75) и (76) будут иметь следующий вид:

$$\{X'\} = pX_1 + \sum_{i=1}^{k-1} \lambda_i (X_{i+1} - pX_1); \quad (77)$$

i	X_i	x_1	x_2	x_3	x_4	x_5	y_i
1	X_1	2	1	2	1	1	23
2	X_2	4	3	3	3	2	48
3	X_3	7	6	10	8	6	117

$$\Phi(X' - X_3) = \sum_{l=1}^n [pX_{1l} + \sum_{i=1}^{k-1} \lambda_i (X_{i+1} - pX_{1l}) - X_{3l}]^2. \quad (78)$$

Для минимизации функции (78) находим производные Φ'_{λ_i} и Φ'_p и приравняем их нулю. В результате получим систему из k линейных уравнений относительно $p, \lambda_1, \dots, \lambda_{k-1}$.

Решение экстраполяционной задачи в случае поворота подпространства проектных ситуаций получим по формуле

$$y_3 = py_1 + \sum_{i=1}^{k-1} \lambda_i (y_{i+1} - py_1). \quad (79)$$

Работоспособность предложенных методов повышения точности экстраполяции, в основу которых положены идеи поворота и сдвига подпространства ситуаций, рассмотрим на следующем примере.

Пусть заданы ситуации X_1 и X_2 и их решения y_1 и y_2 , контрольный (тестовый) базис которых равен $\{3, 5, 4, 1, 3\}$. Требуется определить решение y_3 для новой неизвестной ситуации X_3 . Основные данные, необходимые для решения экстраполяционной задачи, приведены в табл. 2.

Найдем решения несколькими методами и сравним их.

1. Решение по формулам немодифицированного алгоритма многомерной линейной экстраполяции:

$$X'(X_1, X_2) = (2 + 2\lambda, 1 + 2\lambda, 2 + \lambda, 1 + 2\lambda, 1 + \lambda);$$

$$\Phi(X' - X_3) = (2\lambda - 5)^2 + (2\lambda - 5)^2 + (\lambda - 8)^2 + (2\lambda - 7)^2 + (\lambda - 5)^2.$$

Минимизируя Φ , получим $\lambda = 3,34$. Решение для ситуации X_3 находим по экстраполяционной формуле

$$y_3 = y_1 + \lambda(y_2 - y_1) = 107.$$

Погрешность при этом $\delta = 8,5\%$.

2. Решение методом пропорционального сдвига:

$$\{X'\} = p[X_1 + \lambda(X_2 - X_1)];$$

$$\begin{aligned} \Phi(X' - X_3) &= (2p + 2\lambda p - 7)^2 + (p + 2\lambda p - 6)^2 + \\ &+ (2p + \lambda p - 10)^2 + (p + 2\lambda p - 8)^2 - (p + \lambda p - 6)^2. \end{aligned}$$

Минимизируя эту функцию по p и λ , получаем $p = 3,55$; $\lambda = 0,38$. В этом случае для ситуации X_3 решение находится по экстраполяционной формуле модифицированного алгоритма:

$$y_3 = p[y_1 + \lambda(y_2 - y_1)] = 115.$$

Погрешность будет $\delta = 1,6\%$, что значительно меньше, чем погрешность, полученная в первом случае.

3. Решение методом поворота:

$$\{X'\} = pX_1 + \lambda (X_2 - pX_1);$$

$$\Phi(X' - X_3) = [2p(1 - \lambda) + 4\lambda - 7]^2 + [p(1 - \lambda) + 3\lambda - 6]^2 + \\ + [2p(1 - \lambda) + 3\lambda - 10]^2 + [p(1 - \lambda) + 3\lambda - 8]^2 + [p(1 - \lambda) + 2\lambda - 6]^2.$$

Минимизируя Φ по λ и p , получаем $\lambda = 1,38$; $p = -6,8$. Находим решение по экстраполяционной формуле, полученной для модифицированного алгоритма, связанного с поворотом подпространства ситуаций:

$$y_3 = py_1 + \lambda(y_2 - py_1) = 115,5.$$

Погрешность при этом $\delta = 1,3\%$.

Таким образом, предлагаемые модификации алгоритма многомерной линейной экстраполяции, связанные с поворотом и сдвигом подпространства ситуаций, позволяют значительно улучшить результат экстраполяции, что особенно важно при малом числе наблюдений, когда погрешность экстраполяции велика.

Глава II

ПРИМЕНЕНИЕ АЛГОРИТМА МНОГОМЕРНОЙ ЭКСТРАПОЛЯЦИИ ДЛЯ РЕШЕНИЯ ЗАДАЧ ПРОЕКТИРОВАНИЯ

1. ВОССТАНОВЛЕНИЕ НЕИЗВЕСТНЫХ ХАРАКТЕРИСТИК ПРОЕКТИРУЕМОГО ОБЪЕКТА

1. **Модельный пример в трехмерном пространстве.** Решается экстремальная задача эскизного проектирования по показателю качества F (показатель, характеризующий выходные характеристики системы). В распоряжении конструктора имеется две проектные ситуации A_1 (2,0; 3,0; 1,5) и A_2 (4,0; 4,5; 2,7), для которых найдены выходные характеристики системы $F_1 = 34,5$ и $F_2 = 49,9$. Необходимо найти выходную характеристику новой системы с параметрами A_3 (5,5; 6,0; 4,0).

Условия задачи приведены в табл. 3.

Для решения этой задачи воспользуемся методом многомерной экстраполяции.

Из табл. 3 следует, что для ситуации A_3 известно точное значение выходной характеристики, найденное по заданному тест-базису. Это значение будет использовано для оценки погрешности метода экстраполяции.

В соответствии с алгоритмом метода многомерной экстраполяции составляем подпространство известных ситуаций по формуле

$$\{X'\} = A_1 + \lambda(A_2 - A_1).$$

Т а б л и ц а 3

F	A_i	x_1	x_2	x_3
34,5	A_1	2,0	3,0	1,5
49,9	A_2	4,0	4,5	2,7
65,0	A_3	5,5	6,0	4,0

Т а б л и ц а 4

F	A_i	x_1	x_2	x_3	x_4	x_5
19,75	A_1	2,0	1,0	1,5	2,5	1,0
33,00	A_2	3,0	2,5	3,5	3,0	2,0
47,25	A_3	4,0	3,5	5,0	4,5	3,0
85,50	A_4	7,0	8,0	6,0	9,0	5,0

Для данного случая получим покоординатное представление подпространства

$$\{X'\} = \{(2 + 2\lambda), (3 + 1,5\lambda), (1,5 + 1,2\lambda)\}.$$

Далее вводим функцию близости

$$\Phi(X' - A_3) = |X' - A_3|^2.$$

Используя условия задачи, получим количественное представление функции близости в виде

$$\Phi(X' - A_3) = (2\lambda - 3,5)^2 + (1,5\lambda - 3)^2 + (1,2\lambda - 2,5)^2.$$

Минимизируем функцию близости, для чего определим ее производную по параметру λ

$$\Phi_{\lambda}' = 7,69\lambda - 14,5 = 0.$$

Решая полученное уравнение, находим параметр $\lambda = 1,88$.

По формуле (32), которая в данном случае имеет вид

$$F_3 = F_1 + \lambda(F_2 - F_1),$$

получим значение выходной характеристики $F_3^3 = 63,5$.

Сравнивая F_3 и F_3^3 , получим оценку погрешности для данного примера $\delta = 2,35\%$.

Значение ошибки говорит о достаточно высокой эффективности метода многомерной экстраполяции при решении данного класса задач.

2. Модельный пример в пятимерном пространстве. Конструктору известны три проектные ситуации A_1, A_2, A_3 , для которых известны выходные характеристики системы. Требуется найти оптимальную выходную характеристику системы для новой ситуации A_4 . Условия задачи приведены в табл. 4.

Так же как и в предыдущем примере, полагаем, что в ситуации A_4 известно точное значение выходной характеристики системы.

Решим задачу в двух вариантах. Вначале воспользуемся информацией по двум проектным ситуациям A_1 и A_2 .

Алгоритм решения задачи аналогичен рассмотренному ранее, т.е. вначале составляем подпространство известных ситуаций

$$\{X'\} = \{(2 + \lambda), (1 + 1,5\lambda), (1,5\lambda + 2\lambda), (2,5 + 0,5\lambda), (1 + \lambda)\}.$$

Формируем квадратичную функцию близости новой ситуации A_4 к подпространству $\{X'\}$:

$$\Phi(X' - A_4) = (\lambda - 5)^2 + (1,5\lambda - 7)^2 + (2\lambda - 4,5)^2 + (0,5\lambda - 6,5)^2 + (\lambda - 4)^2.$$

Минимизируем квадратичную функцию близости, для чего находим производную по λ и приравняв ее нулю

$$\Phi_{\lambda}' = 8,5\lambda - 31,75 = 0.$$

Решая уравнение, находим параметр $\lambda = 3,735$.

По формуле экстраполяции, отождествляя $\lambda = \mu$, находим выходную характеристику системы в проектной ситуации A_4

$$F_4^3 = F_1 + \lambda(F_2 - F_1) = 69,24.$$

Зная точное значение выходной характеристики системы, оцениваем относительную погрешность экстраполяции $\delta = 19,1\%$.

А теперь решим эту же задачу, используя более полную информацию — проектные ситуации A_1, A_2 и A_3 .

Так же как и в первом случае, составляем подпространство известных ситуаций

$$\{X'\} = \{(2 + \lambda_1 + 2\lambda_2), (1 + 1,5\lambda_1 + 2,5\lambda_2), (1,5 + 2\lambda_1 + 3,5\lambda_2), (2,5 + 0,5\lambda_1 + 2\lambda_2), (1 + \lambda_1 + 2\lambda_2)\}.$$

Вводим квадратичную функцию близости новой ситуации A_4 к подпространству $\{X'\}$

$$\Phi(X' - A_4) = (\lambda_1 + 2\lambda_2 - 5)^2 + (1,5\lambda_1 + 2,5\lambda_2 - 7)^2 + (2\lambda_1 + 3,5\lambda_2 - 4,5)^2 + (0,5\lambda_1 + 2\lambda_2 - 6,5)^2 + (\lambda_1 + 2\lambda_2 - 4)^2.$$

Минимизируем функцию близости, для чего приравняем ее производную по переменным параметрам λ_1 и λ_2 нулю:

$$\Phi_{\lambda_1}' = 8,5\lambda_1 + 15,75\lambda_2 - 31,75 = 0;$$

$$\Phi_{\lambda_2}' = 15,75\lambda_1 + 30,5\lambda_2 - 64,25 = 0.$$

Решая систему уравнений, находим переменные параметры λ_1 и λ_2 , минимизирующие функцию близости:

$$\lambda_1 = 4,39; \lambda_2 = 4,38.$$

Построив подпространство решений по формуле

$$F_4^3 = F_1 + \mu_1(F_2 - F_1) + \mu_2(F_3 - F_1)$$

и вводя тождества $\lambda_1 = \mu_1, \lambda_2 = \mu_2$, находим экстраполированную выходную характеристику системы $F_4^3 = 83,25$.

Используя точное значение модельной задачи, оцениваем относительную погрешность экстраполяции $\delta = 2,7\%$.

Сравнивая результаты экстраполяции характеристик системы A_4 по двум аналогам A_1 и A_2 и по трем аналогам A_1 , A_2 и A_3 , видим, что погрешность экстраполяции снизилась с 19,1 до 2,7 %.

3. **Восстановление неизвестных характеристик реальных объектов.**
В данной части параграфа описано использование метода многомерной экстраполяции для восстановления характеристик реальных объектов: летательных аппаратов (табл. 5), легковых автомобилей (табл. 6), автобусов (табл. 7), транзисторных радиоприемников (табл. 8), стереокомплексов (табл. 9).

В табл. 5 приняты следующие обозначения:

X_1 — масса пустого летательного аппарата, кг;
 X_2 — полная (взлетная) масса, кг;
 X_3 — максимальная высота, м;
 X_4 — максимальная скорость, км/ч;
 X_5 — максимальная скорость у земли, км/ч;
 X_6 — длина летательного аппарата, м;
 X_7 — площадь крыла, м²;
 $F_1 - F_3$ — модели летательного аппарата;
 F_3^3 — модель F_3 , характеристики которой восстановлены методом многогранной экстраполяции.

В табл. 6 — 9 по первым трем образцам восстанавливались характеристики четвертого, т.е. восстанавливались характеристики легкового автомобиля "Москвич-21406", автобуса ЛАЗ-695н, транзисторного радиоприемника "Кварц-407" и стереокомплекса "Мелодия-106А-стерео" (см. соответствующие таблицы).

Работу алгоритма поясним на примере восстановления характеристик

Т а б л и ц а 5

Модель F	X_1	X_2	X_3	X_4	X_5	X_6	X_7
F_1	4027	5200	15600	1114	1060	11,2	22,6
F_2	3507	4491	11500	760	700	12,1	18,8
F_3	3790	4960	15500	1076	1060	10,1	20,6
F_3^3	4025	5197	15580	1113	1059	11,2	22,6
$\delta, \%$	6,2	4,8	0,5	3,4	0,13	11,5	9,6

Т а б л и ц а 6

Модель автомобиля	Рабочий объем цилиндров, см ³	Мощность, кВт	База, мм	Колея, мм	Масса, кг	Скорость, км/ч	Время разгона до 100 км/ч, с
ВАЗ-2101	1198	46	2424	1349	955	140	22
ВАЗ-2106	1568	60	2424	1365	1045	154	17,5
ВАЗ-2121	1568	60	2200	1430	1150	130	25
"Москвич-21406"	1478	51	2400	1270	1080	130	26
$\delta, \%$	0,39	11,64	0,17	7,64	4,28	14,39	13,22

Т а б л и ц а 7

Модель автобуса	Рабочий объем цилиндров, см ³	Мощность, кВт	Частота вращения, об/мин	База, мм	Колея, мм	Масса, кг	Скорость, км/ч
ЛАЗ-3201	4254	86	3200	3600	1812	5070	80
ЛиАЗ-677	6959	135	3200	4150	2100	8000	70
ЛАЗ-4202	10850	135	2600	4370	1860	8600	77
ЛАЗ-695н	5966	113	3200	4190	2116	6850	75
δ, %	0,21	3,26	0,18	8,43	6,12	0,36	1,42

Т а б л и ц а 8

Модель радио-приемника	Число активных элементов	Выходная мощность, Вт	Чувствительность, мВ	Масса, кг	Число сменных элементов		
					L	C	R
"Альпинист-407"	9	0,5	0,7	1,5	16	39	27
"Сокол-405"	12	0,3	0,15	0,8	12	35	36
"Нейва-402"	9	0,15	0,8	0,37	12	28	13
"Кварц-407"	9	0,15	0,6	0,6	15	33	24
δ, %	13,63	15,42	12,97	13,73	2,44	2,34	0,42

Т а б л и ц а 9

Модель стерео-комплексов	Полоса частот, Гц		Чувствительность, мкВ	Выходная мощность, Вт	Потребляемая мощность, Вт	Масса, кг	Цена, руб.
	ДВ, СВ, КВ	УКВ					
"Мелодия-105-стерео"	6300	12500	100	12	70	29	455
"Романтика-112-стерео"	6300	12500	100	20	150	43	990
"Эстония-009-стерео"	3500	16000	150	50	130	17	950
"Мелодия-106А-стерео"	6300	12500	75	20	70	30	706
δ, %	0,2	0,15	13,33	11,73	12,34	18,04	0,85

летательного аппарата. Поскольку каждый летательный аппарат (а их в примере рассмотрено три) описывался семимерным вектором, то в этом случае алгоритм восстановления неизвестных характеристик оказывался достаточно громоздким, и задача решалась на ЭВМ ЕС-1033. Идея алгоритма аналогична идее, рассмотренной ранее: по известным характеристикам двух летательных аппаратов восстанавливались характеристики третьего летательного аппарата (в остальных примерах по

характеристикам трех образцов восстанавливались характеристики четвертого). Решение задачи заканчивалось в тот момент, когда восстанавливались все семь характеристик летательного аппарата. Далее, зная реальные характеристики летательного аппарата, вычисляли ошибки восстановления всех его характеристик.

Анализ результатов, приведенных в табл. 5 — 9 и полученных в начале параграфа, показывает, что метод многомерной экстраполяции можно с успехом применять для решения задач восстановления неизвестных характеристик реальных образцов техники. Так, максимальная ошибка восстановления в приведенных примерах не превысила 18,04 % (табл. 9), что вполне допустимо при проведении технических (ориентировочных) расчетов. В остальных таблицах ошибки восстановления были заметно ниже.

Кроме того, указанные расчеты позволили выявить достаточно тесную связь между ошибками восстановления и выбором образцов-аналогов. При рациональном выборе аналогов ошибки восстановления могут быть значительно снижены.

Из сказанного следует, что метод многомерной экстраполяции можно с успехом применять на стадии эскизного и аванпроектирования.

2. ВОССТАНОВЛЕНИЕ ЧИСЛОВЫХ ТАБЛИЦ

Рассматривается задача восстановления таблиц количественных показателей. Таблица имеет вид прямоугольной матрицы

$$A = \|a_{ij}\|_{N \times M}, \quad (80)$$

где $a_{ij} \in S_{ij}$ ($i = 1, \dots, N; j = 1, \dots, M$); $S = \|S\|_{N \times M}$ — заданная матрица интервалов изменений количественных показателей этой таблицы. Предполагается, что матрица (80) обладает избыточностью, т.е. между элементами таблицы имеются связи, о характере которых априорная информация, вообще говоря, отсутствует. В матрице имеются m пропусков, которые образуют набор значений неизвестных элементов

$$Z = \{z_1, \dots, z_m\}.$$

Для восстановления значений неизвестных элементов можно применить метод многомерной линейной экстраполяции, в котором при отсутствии информации о характере связей между элементами таблицы используется естественная гипотеза о кусочной линейности этих связей. В дальнейшем будем широко применять указанную гипотезу [29].

Рассмотрим сначала простейший случай, когда $m = 1$, т.е. $Z = z$. Тогда в таблице (80) имеется один пропуск на месте элемента a_{kl} . Для восстановления значения z элемента a_{kl} матрицу A делят на три части: X_{kl} , Y_{kl} и V_{kl} , где X_{kl} — матрица размерности $(N - 1) \times (M - 1)$, полученная из матрицы (80) после вычеркивания k -й строки и l -го столбца:

$$\begin{aligned} X_{kl} &= \|a_{ij}\|_{(N-1) \times (M-1)} \text{ при } i \neq k \text{ и } j \neq l; \\ Y_{kl} &= l\text{-й вектор-столбец матрицы (80) без элемента } a_{kl}; \\ V_{kl} &= \|a_{il}\|_{(N-1) \times 1} \text{ при } i \neq k; \end{aligned}$$

$V_{kl} - k$ -я вектор-строка матрицы (80) без элемента a_{kl} : $V_{kl} = \|a_{kj}\|_{1 \times (M-1)}$ при $j \neq l$.

Значение z тогда будет равно оценке неизвестного элемента \hat{a}_{kl} :

$$z = \hat{a}_{kl} = \varphi(X_{kl}, Y_{kl}, V_{kl}), \quad (81)$$

где φ — решающее правило такой оценки.

В качестве решающего правила φ и предлагается применить метод многомерной линейной экстраполяции, позволяющий оценить элемент a_{kl} . Воспользуемся следующей модификацией этого метода. Опорная последовательность образуется матрицей X_{kl} в форме

$$\langle U_i, a_{il} \rangle \quad (i = 1, \dots, k-1, k+1, \dots, N),$$

где $U_i - (M-1)$ -мерный вектор вида

$$U_i = (a_{i1}, \dots, a_{i, l-1}, a_{i, l+1}, \dots, a_{iM}),$$

или в форме

$$\langle W_j, a_{kj} \rangle \quad (j = 1, \dots, l-1, l+1, \dots, M),$$

где $W_j - (N-1)$ -мерный вектор вида

$$W_j = (a_{1j}, \dots, a_{k-1, j}, a_{k+1, j}, \dots, a_{Nj}).$$

Искомая оценка a_{kl} определяется как линейная экстраполяция опорной последовательности по U_k (или W_l). Для этого образуем вектор

$$U = \sum_{i \in L_k} \varphi_i U_i, \quad (82)$$

где $\sum_{i \in L_k} \lambda_i = 1$, а множество L_k состоит из t номеров, подбираемых по признаку близости U_i к U_k в соответствии с выбранной мерой $\rho(U_i, U_k)$, например, евклидовой:

$$\rho(U_i, U_k) = |U_i - U_k|.$$

Как видно, задача отбора U_i , входящих в сумму (82), сводится к выбору t векторов U_i , ближайших к заданному по критерию выбранной меры ρ .

В зависимости от значения t возможны два режима работы алгоритма.

1. Если $t = M$, то вектор параметров

$$\Lambda = (\lambda_1, \dots, \lambda_M)$$

определяется из системы линейных уравнений

$$U_k = \sum_{i \in L_k} \lambda_i U_i; \quad \sum_{i \in L_k} \lambda_i = 1. \quad (83)$$

Это обычная линейная зависимость строк матрицы Λ , которая постулируется для удобства решения задачи. Однако условие $t = M$ далеко не всегда удастся удовлетворить, например, при $N < M$, поэтому особый интерес представляет следующий случай.

2. Если $t < M$, то для определения параметров Λ необходимо ввести дополнительную информацию в виде гипотезы о близости U_k и вектора (82), т.е. в виде той же меры $\rho(U, U_k)$, минимум которой дает возможность определить искомые параметры Λ .

Для этого следует решить задачу минимизации

$$P\left(\sum_{i \in L_k} \lambda_i U_i, U_k\right) \rightarrow \min_{\Lambda \in \Omega} \Rightarrow \Lambda^*, \quad (84)$$

где

$$\Omega: \sum_{i \in L_k} \lambda_i = 1,$$

а Λ^* есть решение задачи минимизации.

Теперь оценка искомого экстраполируемого значения очевидна:

$$\hat{a}_{kl} = \sum_{i \in L_k} \lambda_i^* a_{il}. \quad (85)$$

где λ_i^* ($i \in L_k$) — решение системы (83) или задачи (84). В этом и состоит решающее правило φ в выражении (81).

Далее рассмотрим общий случай: $m \geq 1$. Построим функцию невязки матрицы A и ее поэлементной экстраполяции в виде суммы

$$Q(Z) = \sum_{i,j} f(a_{ij} - \hat{a}_{ij}), \quad (86)$$

где \hat{a}_{ij} — оценка экстраполированного значения a_{ij} изложенным способом (85); f — унимодальная неотрицательная функция, например $f(\cdot) = 1 \cdot |\cdot|$, а в местах соответствующих пропусков матрицы A стоят параметры вектора Z .

Тогда задача восстановления таблицы представляется в виде задачи оптимизации: необходимо минимизировать суммарную невязку (86) экстраполяции всех значений элементов таблицы подбором значений неизвестных элементов, т.е. решить задачу

$$Q(Z) \rightarrow \min_{Z \in S} \Rightarrow Z^*, \quad (87)$$

где $Z^* = (z_1^*, \dots, z_m^*)$ — решение этой задачи — оценки значений неизвестных элементов; S — заданная ранее матрица интервалов изменения элементной таблицы A . Выбор метода оптимизации для решения задачи (87) зависит, очевидно, от характера функции (86).

Для анализа поведения функции невязки (86) решали следующие модельные задачи. Модель представляла собой матрицу (80) с заданной избыточностью. Избыточность моделировалась простейшей линейной зависимостью строк матрицы (80) следующим образом: первые $l < N$ строк матрицы были независимы, их заполняли целыми случайными числами с равномерным распределением в интервале $[0; 9]$, а остальные $N - l$ строк представляли собой линейные комбинации независимых

строк, т.е. $a_{ij} = \sum_{k=1}^l \alpha_{ik} a_{kj}$ для $i = l + 1, \dots, N; j = 1, \dots, M$, где α_{ik} — целые случайные числа с равномерным распределением в интервале $[0; 9]$. Ранг r такой матрицы, очевидно, равен числу линейно-независимых строк, т.е. $r = l$.

Пример 1. $N = 5$; $M = 4$; $l = 2$; $m = 1$. Функция невязки (86) для данного примера представлена на рис. 7. Хорошо видно, что она имеет экстремальный характер, причем ее минимум соответствует истинному значению восстанавливаемого элемента. Любой локальный метод однопараметрической оптимизации этой функции обеспечит точное восстановление утраченного элемента матрицы.

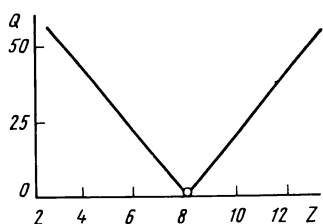
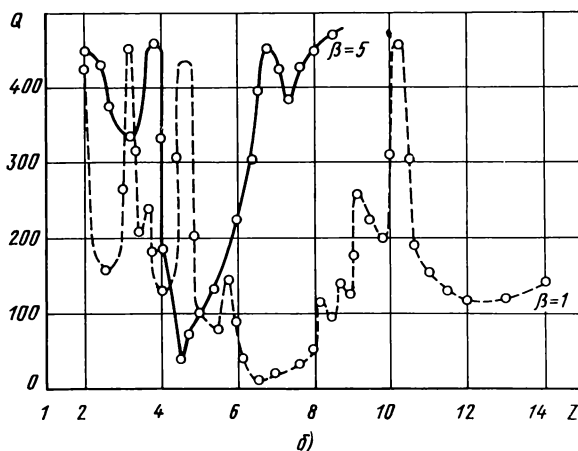
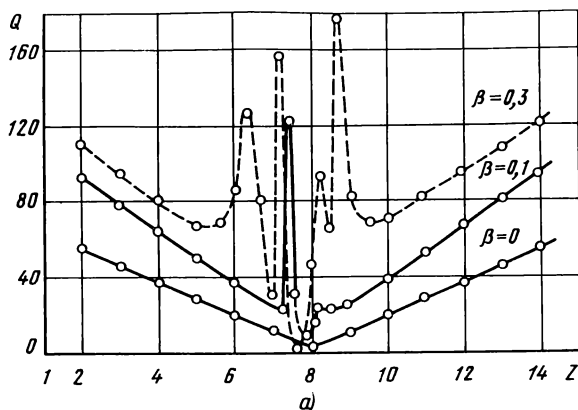


Рис. 7. Функция невязки линейно-зависимой таблицы ($m = 1$)

Рис. 8. Функция невязки линейно-зависимой таблицы со стохастическим возмущением:

a — при малом коэффициенте стохастичности β ; b — при большом β



Нетрудно заметить, что для точного восстановления одного элемента такой линейно-избыточной матрицы необходимо, чтобы

$$r = \min (N - 1; M - 1).$$

Произвольный характер восстанавливаемой матрицы моделировали, вводя элемент стохастичности. Для этого ко всем значениям элементов a_{ij} линейно-избыточной матрицы добавляли стохастическую составляющую

$$a_{ij}' = a_{ij} + \beta \epsilon_k,$$

где a_{ij}' — элемент моделируемой матрицы; a_{ij} — элемент линейно-избыточной матрицы; β — коэффициент стохастичности; ϵ_k — независимые случайные числа, равномерно распределенные в интервале $[-1, 1]$, $k = NM - 1$.

Вид функции (86) для различных значений коэффициента стохастичности β приведен на рис. 8. В отличие от первой модели функция невязки здесь многоэкстремальна, ее глобальный минимум и решает задачу восстановления. Ниже приведены положения экстремумов функции невязки для различных значений коэффициента стохастичности β .

$\beta \dots$	0	0,1	0,3	1	2	3	4	5
$Z^* \dots$	8,0	7,8	7,6	6,5	5,6	5,2	4,9	4,6

Видно, что значительное случайное изменение β ($\beta = 3 - 5$) незначительно изменяет положение глобального экстремума функции невязки. Это свидетельствует об эффективности предложенного способа восстановления пропусков в таблице.

Пример 2. Теперь рассмотрим случай восстановления матрицы с двумя ($m=2$) пропущенными элементами. На рис. 9 представлены линии равного уровня функции невязки для линейно-зависимой модели матрицы А, но при $m = 2$. Функция $Q(z_1, z_2)$ имеет нулевое минимальное значение при аргументах, совпадающих с точными значениями восстанавливаемых элементов.

Как видно, задача унимодальна, и экстремум в данном случае находят любым локальным методом поисковой оптимизации.

На рис. 10, где показаны линии равного уровня функции невязки для той же таблицы, но "зашумленной" стохастической добавкой при $\beta = 0,1$, видно, что минимальное значение функции невязки не равно нулю, но ее оптимум близок к истинному. В этом случае задача также многоэкстремальна, и для ее решения целесообразно применять глобальные методы случайного поиска [37].

Таким образом, применение метода многомерной линейной экстраполяции в сочетании с процедурой параметрической оптимизации дает возможность достаточно точно восстанавливать элементы избыточной матрицы.

Для иллюстрации процесса восстановления пропущенных значений матрицы были проведены два цикла экспериментального поиска со стохастическим сглаживанием (рис. 11). Сглаживание проводилось на базе четырех случайных измерений:

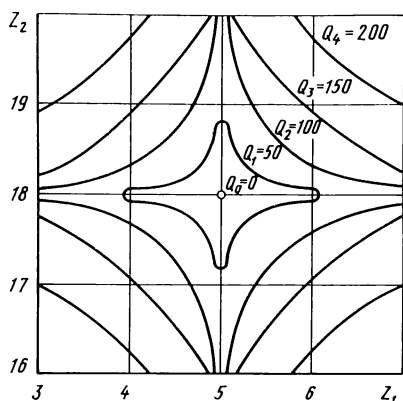


Рис. 9. Линии равного уровня функции невязки линейно-зависимой модели матрицы при $m = 2$

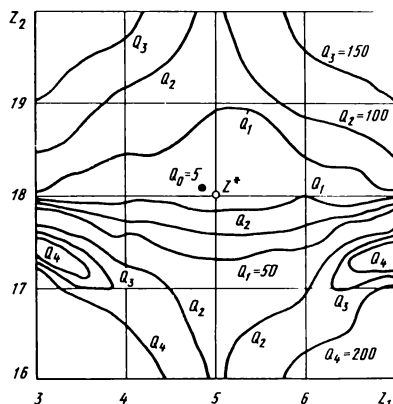


Рис. 10. Линии равного уровня функции невязки линейно-избыточной модели матрицы с коэффициентом стохастичности $\beta = 0,1$

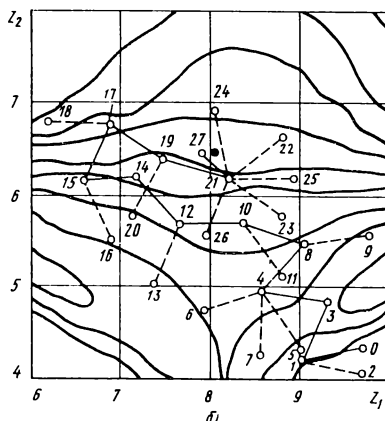
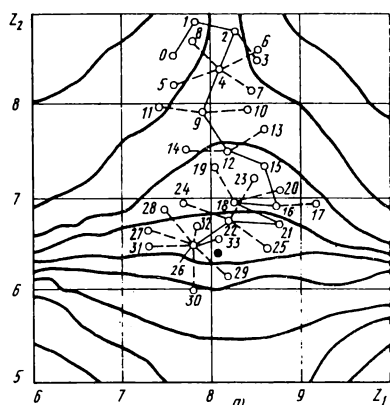


Рис. 11. Траектории случайного поиска:

$a - Z_0 = (7,6; 8,5)$; $a = 0,5$;
 $g = 0,2$; $Z_{3,3} = (8,05; 6,52)$; $\theta - Z_0 =$
 $= (9,7; 4,4)$; $a = 0,75$; $g = 0,3$;
 $Z_{2,7} = (7,95; 6,32)$

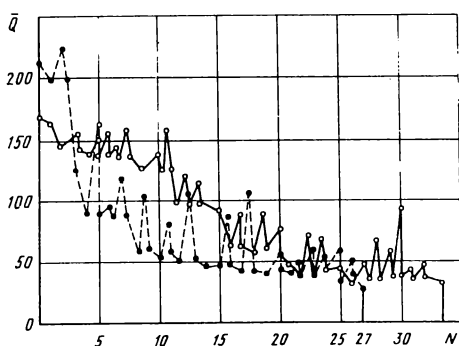


Рис. 12. Поведение сглаженного значения минимизируемой функции при восстановлении таблицы для двух начальных условий:

— процесс a ; --- процесс b (см. рис. 11)

$$Q(Z) = \frac{1}{4} \sum_{i=1}^4 Q(Z + g \xi_i), \quad (88)$$

где g — значение области сглаживания; ξ_i — i -я реализация единичного случайного вектора, равномерно распределенного в пространстве параметров $\{Z\}$. В качестве алгоритма поиска был использован адаптивный случайный поиск с "возвратом":

$$\Delta Z_N = \begin{cases} a \xi_N & \text{при } \Delta \bar{Q}_{N-1} < 0; \\ -\Delta Z_{N-1} + a \xi_N & \text{при } \Delta \bar{Q}_{N-1} \geq 0, \end{cases}$$

где значение шага a уменьшалось в 2 раза после пяти неудачных шагов (и возвратов).

На рис. 11 показаны две траектории поиска минимума стохастически сглаженной функции (88); штриховыми линиями показаны неудачные шаги. Хорошо видно, что в обоих случаях поиск вывел в зону экстремума, причем именно здесь был уменьшен рабочий шаг.

На рис. 12 наглядно видна сходимость процесса поиска.

Таким образом, применение глобального случайного поиска позволяет эффективно восстанавливать утраченные элементы таблицы.

3. ОЦЕНКА ЧИСЛОВЫХ ХАРАКТЕРИСТИК ФУНКЦИИ

Рассмотрим задачу определения числовых характеристик функции, заданной на конечном множестве точек интервала наблюдения. Характеристиками могут быть коэффициенты разложения функции в ряд по заданным функциям, ее производные в некоторых точках интервала и т.п. В данном параграфе в качестве числовых характеристик выбраны коэффициенты аппроксимирующего полинома [28].

Пусть в точках t_i ($i = \overline{1, m}$) интервала наблюдения $[t_1; t_m]$ заданы значения некоторой функции $F(t)$. Требуется аппроксимировать ее полиномом заданной степени $P(t) = \sum_{i=0}^n x_i t^i$, если из предыдущего опыта известно, что функции $F_j(t)$, определенным образом близкие заданной, наилучшим образом аппроксимируются полиномами $P_j(t)$ ($j = \overline{1, k}$) степени n . Функции $F(t)$ и $F_j(t)$ предполагаются близкими в том смысле, что они представляют собой аналогичные процессы, протекающие в таких же условиях. При малой размерности полинома по сравнению с числом точек, т.е. при $m \geq n + 1$, задача решается известными методами равномерных или средних квадратических приближений, причем для этого достаточно наблюдать только функцию $F(t)$.

При $m < n + 1$ через m точек, в которых заданы значения функции, можно провести бесчисленное множество полиномов степени n . Для устранения получающейся многозначности используем априорные сведения о функциях $F_j(t)$ и соответствующих им полиномах $P_j(t)$, для чего применим метод многомерной экстраполяции.

Для решения поставленной задачи методом многомерной экстраполяции введем следующие обозначения:

$A_j = (a_{1j}, a_{2j}, \dots, a_{mj})$ — m -мерный вектор, компонентами которого являются значения функции $F_j(t)$ в точках t_i интервала наблюдения;

$X_j = (x_{0j}, x_{1j}, \dots, x_{nj})$ — $(n + 1)$ -мерный вектор, компонентами которого служат коэффициенты полинома $P_j(t)$, аппроксимирующего функцию $F(t)$ на интервале наблюдения;

$A = (a_1, a_2, \dots, a_m)$ — вектор заданных значений функции $F(t)$ в определенных точках интервала наблюдения $a_i = F(t_i)$;

$X = (x_0, x_1, \dots, x_n)$ — вектор неизвестных коэффициентов полинома $P(t) = \sum_{i=0}^n x_i t^i$;

X^* — оценка вектора X .

В принятых обозначениях априорные сведения о функциях $F_j(t)$ и соответствующих им полиномах $P_j(t)$ можно представить в виде обучающей последовательности. Тогда оценка X^* вектора неизвестных коэффициентов полинома степени n , аппроксимирующего на интервале наблюдения функцию $F(t)$, определяется по формуле экстраполяции (32).

Запишем выражение (31) в скалярной форме:

$$\left. \begin{aligned} x_0^* &= x_{01} + \sum_{j=1}^{k-1} \lambda_j^* (x_{0,j+1} - x_{01}); \\ x_1^* &= x_{11} + \sum_{j=1}^{k-1} \lambda_j^* (x_{1,j+1} - x_{11}); \\ &\dots\dots\dots \\ x_n^* &= x_{n1} + \sum_{j=1}^{k-1} \lambda_j^* (x_{n,j+1} - x_{n1}). \end{aligned} \right\} \quad (89)$$

Умножим первое уравнение системы (89) на единицу, второе — на t_1 , третье — на t_1^2 и т.д. Полученные выражения сложим. Выполним аналогичные операции над системой (89) для моментов времени t_2, \dots, t_m и предположим, что функции $F_j(t)$ являются полиномами n -й степени:

$$\left. \begin{aligned} a_{ij} &= x_{0j} + x_{1j} t_i + x_{2j} t_i^2 + \dots + x_{nj} t_i^n; \\ a_i^* &= x_0^* + x_1^* t_i + x_2^* t_i^2 + \dots + x_n^* t_i^n. \end{aligned} \right\} \quad (90)$$

Тогда

$$\left. \begin{aligned} a_1^* &= a_{11} + \sum_{j=1}^{k-1} \lambda_j^* (a_{1,j+1} - a_{11}); \\ &\dots\dots\dots \\ a_m^* &= a_{m1} + \sum_{j=1}^{k-1} \lambda_j^* (a_{m,j+1} - a_{m1}). \end{aligned} \right\} \quad (91)$$

Как видно, полученные выражения являются скалярной записью формулы (31). Следовательно, в рассмотренном случае гипотеза о линейности преобразования пространства ситуаций $\{A\}$ в пространстве характеристик функций $\{X\}$ справедлива.

Это обстоятельство позволяет сделать следующий вывод: если функции $F_j(t)$ и $F(t)$ являются полиномами n -й степени и если матрицы систем линейно независимых уравнений (89) и (91) квадратны, т.е. выполнены условия

$$\left. \begin{aligned} k &= m + 1; \\ m &= n + 1; \end{aligned} \right\} \quad (92)$$

то методом многомерной экстраполяции будут найдены точные значения коэффициентов полинома $P(t)$. При этом характер априорной информации не оказывает никакого влияния на решение, т.е. функции $F_j(t)$ могут как угодно сильно отличаться от функции $F(t)$.

Действительно, по методу многомерной экстраполяции искомые коэффициенты полинома $x_0^*, x_1^*, \dots, x_n^*$ можно найти из системы уравнений (89) после подстановки в нее коэффициентов $\lambda_1^*, \lambda_2^*, \dots, \lambda_{k-1}^*$, определяемых из системы (91).

Объединим теперь системы (89) и (90):

$$\left. \begin{aligned} a_1^* &= x_0^* + x_1^* t_1 + x_2^* t_1^2 + \dots + x_n^* t_1^n; \\ &\dots\dots\dots \\ a_m^* &= x_0^* + x_1^* t_m + x_2^* t_m^2 + \dots + x_n^* t_m^n; \\ x_0^* &= x_{01} + \sum_{j=1}^{k-1} \lambda_j^* (x_{0,j+1} - x_{01}); \\ x_1^* &= x_{11} + \sum_{j=1}^{k-1} \lambda_j^* (x_{1,j+1} - x_{11}); \\ &\dots\dots\dots \\ x_n^* &= x_{n1} + \sum_{j=1}^{k-1} \lambda_j^* (x_{n,j+1} - x_{n1}). \end{aligned} \right\} \quad (93)$$

В полученной системе число уравнений $(m + n + 1)$ равно по условию (92) числу $(n + k)$ неизвестных $x_0^*, x_1^*, x_2^*, \dots, x_n^*, \lambda_1^*, \dots, \lambda_k^* - 1$. Поскольку матрицы систем (89), (91) и (93) квадратны, а системы (91) и (93) связаны линейным преобразованием, то решение, найденное из (93), совпадает с решением, полученным по методу многомерной экстраполяции. С другой стороны, из системы (93) видно, что искомые коэффициенты полинома можно однозначно определить только из первых m уравнений, не используя априорную информацию, заключенную в остальных $n + 1$ уравнениях. Следовательно, полином $P(t)$, найденный методом многомерной экстраполяции, совпадает с точным полиномом, построенным только из условия прохождения функции через заданные m точек.

Метод многомерной экстраполяции позволяет решать поставленную задачу для случаев

$$\left. \begin{array}{l} k = m + 1; \\ m < n + 1 \end{array} \right\} \quad (94)$$

или

$$\left. \begin{array}{l} k < m + 1; \\ m < n + 1, \end{array} \right\} \quad (95)$$

когда априорной информации оказывается недостаточно для построения линейной модели пространства ситуации или когда размерности пространства ситуаций и пространства параметров различны. Случай $k < m + 1$; $m \geq n + 1$ здесь не рассматривается, так как при $m \geq n + 1$ аппроксимирующий полином можно построить известными методами равномерных или средних квадратических приближений. Проанализируем результаты, которые получаются, если в указанных случаях воспользоваться методом многомерной экстраполяции.

В случае условий (94) из системы (91) или непосредственно из системы (93) находят единственное решение X^* , так как число уравнений $(m + n + 1)$ здесь равно числу неизвестных $(n + k)$. Найденный полином будет проходить строго через заданные точки, но, согласно доказанному выше утверждению, не совпадет с истинным полиномом $P(t)$. Действительно, при $m < n + 1$ матрица системы (89) прямоугольная, и коэффициенты искомого полинома однозначно не могут быть определены только из первых m уравнений системы (93).

Итак, в случае условий (94) из множества полиномов, проходящих через заданные m точек, будет выбран единственный полином, коэффициенты которого связаны линейными зависимостями [последние $n + 1$ уравнений системы (93)] с коэффициентами полиномов обучающей последовательности.

В случае условий (95) по методу многомерной экстраполяции находят такие коэффициенты λ_j^* , при которых среднее квадратическое отклонение полинома $P(t)$ от функции $F(t)$ минимально.

Для экспериментальной проверки определения точности восстановления функции была составлена программа для ЭВМ, реализующая алгоритм многомерной экстраполяции. Исходными данными в программе

являются числовые значения компонент векторов обучающей последовательности и вектора новой ситуации. Программа автоматически настраивается на указанные значения размерности вектора ситуации m , размерности вектора параметров системы n и длины обучающей последовательности k .

С помощью этой программы был проведен следующий эксперимент. Находили оценки коэффициентов полинома седьмого порядка ($n = 7$), аппроксимирующего функцию $F(t)$, которая задавалась своими значениями в $m \leq n$ точках интервала наблюдения. Функции $F_j(t)$ ($j = \overline{1, k}$), определяющие известные ситуации обучающей последовательности, и функция $F(t)$, определяющая новую ситуацию, были выбраны в виде полиномов седьмого порядка. Основное внимание в эксперименте уделялось исследованию случаев при условиях (94) и (95). Поэтому число точек, в которых считались известными значения функций $F(t)$ и $F_j(t)$, а также число полиномов обучающей последовательности варьировалось в пределах

$$4 \leq m \leq 7;$$

$$4 \leq k \leq 7.$$

При этом соблюдалось условие $k \leq m + 1$. Коэффициенты полиномов обучающей последовательности, по которым формировались известные ситуации, выбирались случайно (с помощью датчика случайных чисел) из заданного гиперкуба. На рис. 13 тонкими линиями показаны четыре полинома обучающей последовательности $F_1(t) - F_4(t)$, а жирной линией — функция $F(t)$. Очевидно, вектор коэффициентов аппроксимирующего полинома, который получается в результате решения задачи, является случайным вектором. Для определения вероятностных характеристик (математического ожидания и дисперсии) этого вектора проводили серию из трехсот испытаний при каждом сочетании m и k . Оказалось, что математическое ожидание полинома $P(t)$ (в заданных точках

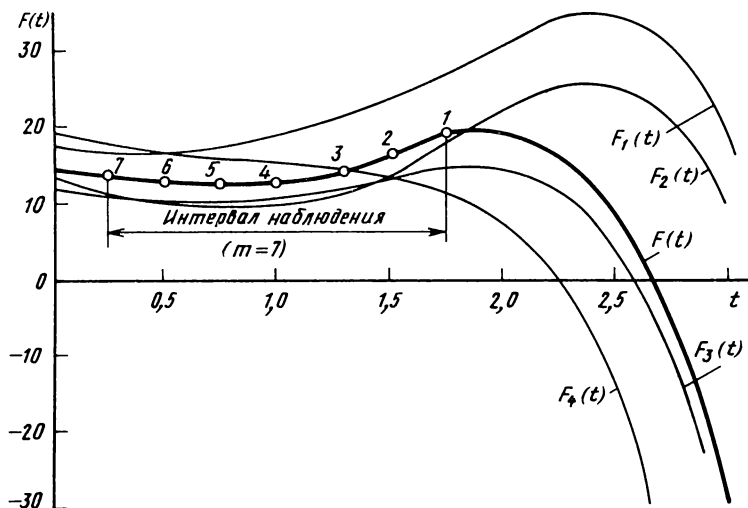


Рис. 13. Функция $F(t)$ и полиномы обучающей последовательности

интервала наблюдения и вне его) практически совпадает с заданной функцией $F(t)$. Таким образом, метод многомерной экстраполяции позволяет найти несмещенную оценку для вектора коэффициентов аппроксимирующего полинома.

Среднее квадратическое отклонение σ полинома $P(t)$ от функции $F(t)$ в заданных точках интервала наблюдения по значению очень мало и уменьшается по мере приближения k к m , а при $k = m + 1$ становится равным нулю. Однако из равенства σ нулю в данном примере еще не следует, что коэффициенты аппроксимирующего полинома $P(t)$ совпали с коэффициентами полинома $F(t)$. В действительности этот факт говорит о том, что из бесчисленного множества полиномов n -й степени, проходящих через $m \leq n$ точек, мы выбрали один, наложив дополнительные условия на коэффициенты искомого полинома. Поэтому отклонение полинома $P(t)$ от функции $F(t)$ характеризовалось по значению среднего квадратического отклонения в точках вне интервала наблюдения, т.е. по качеству предсказания функции $F(t)$ с помощью полинома $P(t)$.

На рис. 14, а, б показана ошибка предсказания функции $F(t)$ при разной длине обучающей последовательности и разном числе точек, взятых на интервале наблюдения. Ширина каждой трубки по оси ординат составляет 3σ . Таким образом, при любой выборке обучающей последовательности длины k предсказанные значения функции $F(t)$ с вероятностью 99,7 % окажутся внутри соответствующей трубки.

Анализ экспериментальных кривых, характеризующих влияние длины обучающей последовательности и влияние числа точек, взятых на интервале наблюдения, на ошибку предсказания функции $F(t)$ (рис. 15, 16) показывает, что при $m = \text{const}$ увеличение длины обучающей последовательности до $k = m + 1$ приводит к уменьшению ошибки предсказания функции (рис. 15). Это следует из того, что увеличение числа известных ситуаций до указанного предела приводит к уменьшению расстояния между вектором новой ситуации A и ближайшим к нему вектором A' , принадлежащим подпространству известных ситуаций, а значит, к уменьшению ошибки, которую мы допускаем при отождествлении векторов A и A' .

Дальнейшее увеличение k может привести к уменьшению ошибки предсказания функции в том случае, если будут предъявлены ситуации, более близкие к новой ситуации, чем некоторые из входящих в число $m + 1$.

Увеличение m при постоянной длине обучающей последовательности может привести как к увеличению, так и к уменьшению ошибки предсказания функции, поскольку, с одной стороны, увеличение здесь означает повышение размерности пространства ситуаций при постоянной размерности подпространства известных ситуаций (что приводит формально к увеличению расстояния между векторами A и A'), а с другой стороны, в этом случае вектор A отождествляется уже с некоторым другим вектором A'' .

Увеличение m приводит к уменьшению ошибки предсказания функции только при одновременном увеличении длины обучающей последовательности. Это иллюстрируется рис. 16.

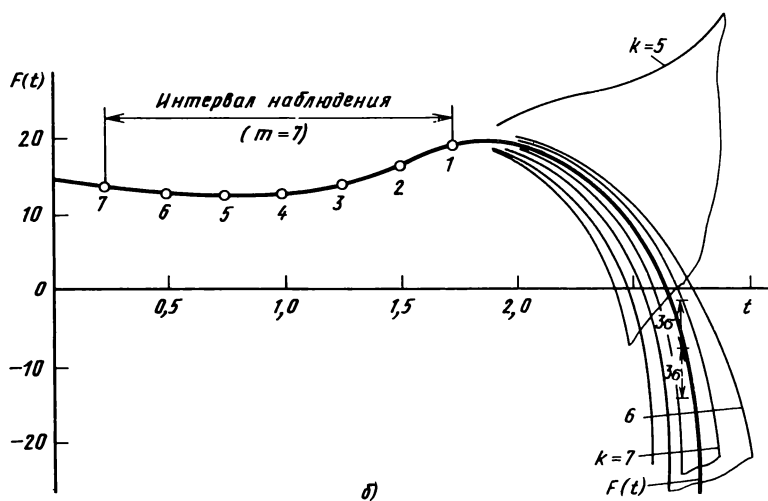
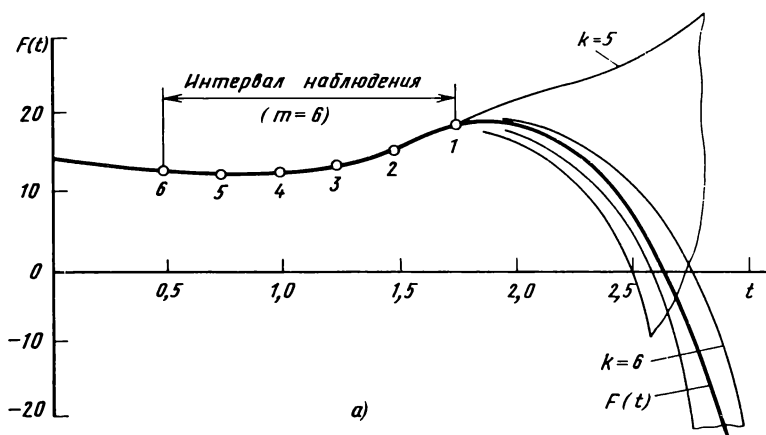


Рис. 14. Ошибка предсказания функции: а – при $m = 6$; б – при $m = 7$

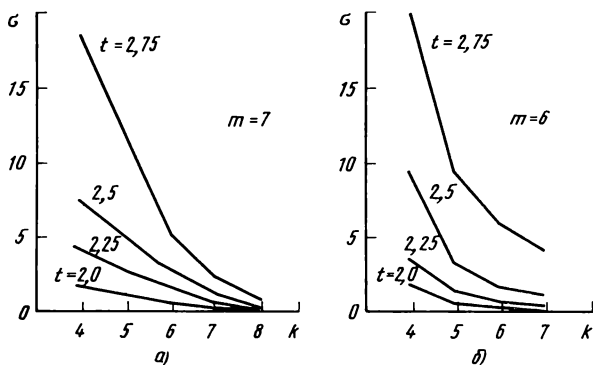


Рис. 15. Влияние длины обучающей последовательности на ошибку предсказания функции: а – при $m = 7$; б – при $m = 6$

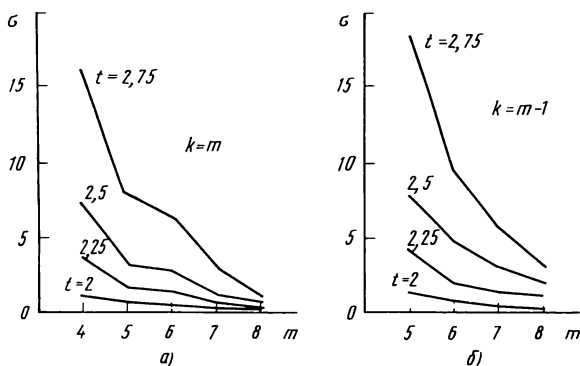


Рис. 16. Влияние числа точек внутри интервала наблюдения на ошибку предсказания функции: а – при $k = m$; б – при $k = m - 1$

В заключение следует отметить, что использование априорной информации в виде обучающей последовательности позволяет уменьшить ошибки предсказания функции при малом интервале наблюдения этой функции. Предсказание поведения функции $F(t)$ без учета априорной информации в рассмотренном примере можно осуществить построением аппроксимирующих полиномов до шестой степени включительно. Нетрудно убедиться, что ошибка предсказания функции с помощью этих полиномов будет гораздо больше, чем по методу многомерной экстраполяции.

4. ИСПОЛЬЗОВАНИЕ МЕТОДА ЭКСТРАПОЛЯЦИИ ДЛЯ РЕШЕНИЯ ЗАДАЧ РАСПОЗНАВАНИЯ ОБРАЗОВ

Существующие методы распознавания образов достаточно четко подразделяются на интегральные и локальные. К первому классу относятся методы, основанные на использовании априорных сведений о функции распределения объектов в пространстве признаков, ко второму классу – методы без использования такой информации [5, 7]. Отличительной особенностью локальных методов является построение решающего правила не по всей обучающей последовательности, а лишь по ее части, оказавшейся в некоторой окрестности подлежащего классификации объекта. При большом объеме обучающей последовательности локальный подход может довольно серьезно конкурировать с параметрическими (интегральными) процедурами распознавания.

В данном параграфе рассмотрим локальный алгоритм распознавания, в основу которого положен метод многомерной линейной экстраполяции [30].

Алгоритм распознавания, основанный на многомерной линейной экстраполяции, заключается в следующем.

Пусть заданы N объектов обучающей последовательности

$$X_i \in E^m, y_i \in \{\omega_1, \dots, \omega_S\}, i = \overline{1, N},$$

где m – размерность пространства признаков; $\omega_1, \dots, \omega_S$ – распознаваемые классы; S – число классов.

Для простоты рассмотрим два класса ($S = 2$), т.е. $y_i \in \{0, 1\}$.

На основе априорной информации, $\langle X_i, y_i \rangle$, $i = \overline{1, N}$, должны приниматься решения о принадлежности к одному из классов любого нового объекта $X_t \in E^m$, $t \neq \overline{1, N}$, подлежащего классификации, т.е. необходимо оценивать y_t .

В соответствии с алгоритмом многомерной линейной экстраполяции значение y_t представляет собой непрерывную оценку [30]

$$\hat{y}_t = \sum_{i=1}^n \lambda_i^* y_i, \quad (96)$$

которую для случая распознавания двух классов естественно дискретизировать введением порога δ :

$$y_t = \begin{cases} 1, & \text{если } \hat{y}_t \geq \delta; \\ 0, & \text{если } \hat{y}_t < \delta. \end{cases} \quad (97)$$

В рассматриваемом алгоритме факторами, влияющими на качество распознавания, являются метрика $\rho(x_i, x_j)$, величина $n \leq m + 1$, порог $0 < \delta < 1$. Указанные факторы вообще должны быть заданы. Однако они могут быть определены и адаптивным путем. Для этого требуется определить функционал Q , характеризующий эффективность классификации на части обучающей последовательности в некоторой окрестности дискриминируемого объекта. Функционал Q будет зависеть от $\rho(\cdot, \cdot)$, n , δ . Экстремизацией Q по указанным факторам находят их оптимальные значения для данной окрестности

$$Q(\rho, n, \delta) \xrightarrow{\rho, n, \delta} \text{extr} \Rightarrow \rho^*, n^*, \delta^*.$$

Необходимо отметить, что как оптимальная метрика ρ^* , так и величины n^* и δ^* являются локальными характеристиками пространства признаков, имеющими в разных точках пространства различные значения.

В реальных задачах классы объектов большей частью не имеют строго определенных критериев принадлежности. Поэтому представляется весьма перспективным подход к анализу сложных систем с учетом нечеткости разделения объектов на классы, предложенный Заде. Нечеткий (размытый) класс Z в области определения объектов $\{X\}$ характеризуется функцией принадлежности к этому классу $\mu_Z(X)$, которая каждому элементу X_i множества ставит в соответствие число $\mu_Z(X_i)$ на отрезке $[0, 1]$. Величина $\mu_Z(X_i)$ отражает степень принадлежности элемента X_i к классу Z [10]. Таким образом, при решении задач с размытыми классами проблема распознавания сводится к определению степени принадлежности классифицируемого объекта X_t к каждому из классов, т.е. к определению значений функции принадлежности $\mu_Z(X_t)$, где $Z = \omega_1, \dots, \omega_S$.

Рассмотрим примеры применения предложенного алгоритма в случае четкого задания обучающей последовательности.

Пусть объекты обучающей последовательности распределены равномерно и регулярно ($N = 25$) по площади единичного квадрата ($m = 2$). Разделяющая два класса поверхность представляет собой кусочно-линейную кривую, делящую площадь квадрата на две равные части:

$$y = \begin{cases} 1, & \text{если } (x_1 + 0,5x_2 - 0,5 < 0) \vee (x_1 - 0,5x_2 - 0,5 > 0); \\ 0 & - \text{в противном случае.} \end{cases}$$

В качестве метрики выберем евклидово расстояние $\rho(X, X_i) = |X - X_i|$, порог $\delta = 0,5$. Критерием качества классификации естественно считать вероятность ошибки классификации p .

Область ошибочной классификации в данном примере довольно легко определяется построением разделяющей кривой, соответствующей решающему правилу (97). Ввиду равномерности распределения объектов вероятность ошибки классификации определяется площадью области ошибочной классификации [30]. Для $n = 2$ значение $p = 0,1$, а для $n = 3$ имеем $p = 0,06$.

Определим теперь вероятность ошибки классификации при распознавании одним из наиболее изученных локальных методов — методом Фикса—Ходжеса [5].

Согласно этому методу рассматривается нечетное число k ближайших к X_i объектов из обучающей последовательности. Объект X_i относится к первому классу, если среди них преобладают объекты первого класса, и ко второму классу в противном случае.

Для данного примера при $k = 3$ получаем $p = 0,08$. Сравнение результатов показывает, что при одной и той же априорной информации $k = n = 3$ метод многомерной экстраполяции оказывается эффективнее метода Фикса — Ходжеса.

Сравним эффективность тех же методов при случайном формировании обучающей последовательности для рассмотренной модельной задачи. В эксперименте объекты моделировались датчиком случайных чисел с равномерным законом распределения в интервале $[0, 1]$. Получены следующие вероятности ошибки классификации: для экстраполяционного алгоритма $\hat{p} = 0,138 (\pm 0,028)$; для метода Фикса — Ходжеса $\hat{p} = 0,162 (\pm 0,034)$. Результаты усреднены по 10 различным обучающим последовательностям ($N = 25$) при одной и той же экзаменационной последовательности (50 объектов). В скобках указаны доверительные интервалы вероятности. Анализ этих результатов показывает, что и в этом случае экстраполяционный алгоритм оказывается эффективнее метода Фикса — Ходжеса.

Как уже отмечалось выше, процедуру определения оптимальных значений факторов, влияющих на качество классификации, можно включить в алгоритм. Тогда для каждого распознаваемого объекта X_i предварительно будет определяться такая совокупность факторов, которой соответствует наилучшее локальное решающее правило для окрестности этого объекта. Необходимо отметить, что наилучшим для X_i может оказаться правило, построенное не обязательно по ближайшим к нему векторам. Поэтому модифицированный алгоритм предусматривает определение как оптимальных значений n^* и δ^* , так и набора этих n^* векторов для каждого X_i . Определение метрики $\rho(\dots)$ в данном алгоритме не предусмотрено, хотя в принципе возможно. Для определения наилучшего локального решающего правила необходимо найти локальную обучающую последовательность из k ближайших в соответ-

ствии с выбранной метрикой. Тогда модифицированный алгоритм состоит из следующих этапов.

1. Из локальной обучающей последовательности формируются различные комбинации по n объектов ($n = 2, 3, \dots, m+1$). Каждая комбинация (обозначим ее $\{X\}_n$) определяет локальное решающее правило. Значение ошибки классификации оставшихся $k - n$ объектов служит оценкой качества этого правила:

$$\hat{P} = \frac{1}{k - n} \sum_{i=1}^{k-n} \varphi_i(X_i),$$

где

$$\varphi_i(X_i) = \begin{cases} 1, & \text{если } X_i \text{ классифицирован неправильно;} \\ 0 & \text{в противном случае.} \end{cases}$$

2. Определяется такая комбинация векторов $\{X\}_{n^*}$, которая соответствует наименьшему значению \hat{P} :

$$\hat{P}(\{X\}_n) \rightarrow \min_{\{X\} \in C_k^n} \Rightarrow \{X\}_{n^*},$$

где $\{X\}$ — комбинация из n объектов ($n = \overline{2, m+1}$); C_k^n — множество всех возможных комбинаций $\{X\}_n$; $\{X\}_{n^*}$ — комбинация векторов, определяющая локальное решающее правило для окрестности X_i .

3. При $\hat{P}^* = \hat{P}(\{X\}_{n^*}) \neq 0$ возможно дальнейшее улучшение решающего правила посредством определения оптимального для данной окрестности значения порога δ :

$$\hat{P}(\delta) \rightarrow \min_{0 < \delta < 1} \Rightarrow \delta^*.$$

4. Полученные $\{X\}_{n^*}$ и δ^* используются для оценки \hat{y}_i и соответственно для классификации объекта X_i с учетом (97).

Процедура определения значения k в алгоритме не предусмотрена. Значение k можно выбирать исходя из объема обучающей последовательности N .

Работоспособность модифицированного алгоритма проверялась на рассмотренной ранее модельной задаче. Зависимость вероятности ошибки классификации P от величины k исследовалась тремя методами:

- методом Фикса — Ходжеса;
- методом экстраполяции без подстройки порога;
- методом экстраполяции с подстройкой порога.

Усредненные по пяти различным обучающим последовательностям ($N = 25$) результаты эксперимента приведены на рис. 17. Экзаменационная последовательность состояла из 50 объектов. Из рисунка видно, что для каждого метода существует оптимальное значение k^* . Причем подстройка порога δ с ростом k повышает эффективность алгоритма.

Покажем эффективность модифицированного алгоритма распознавания на примере решения задачи прогнозирования статистической устойчивости динамической системы.

Пусть состояние динамической системы характеризуется семью параметрами ($m = 7$). При этом система может находиться либо в нор-

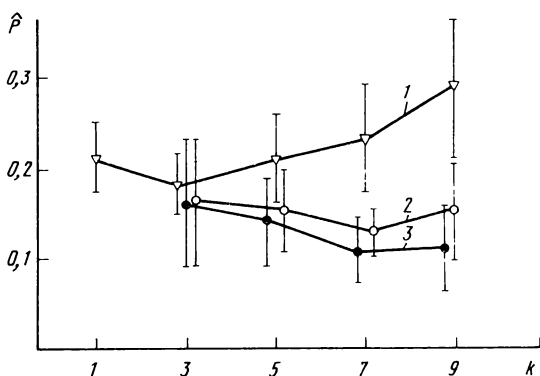


Рис. 17. Зависимость вероятности ошибки классификации от величины k : 1 – метод Фикса – Ходжеса; 2 – метод экстраполяции без подстройки порога; 3 – метод экстраполяции с подстройкой порога

мальном ("норма"), либо в аварийном ("авария") режимах. Задача заключается в прогнозировании режима системы на основе информации о ее параметрах.

Для решения задачи были случайным образом отобраны 120 состояний системы (60 – "норма", 60 – "авария"). Для модифицированного алгоритма с целью сокращения затрат машинного времени n было принято равным 2 и 3 (при $k = 5$). Распознавание проводилось в каждой из заданных 120 точек по 119 остальным. Полученные результаты – для немодифицированного алгоритма $\hat{P} = 0,283$, для модифицированного алгоритма $\hat{P} = 0,192$ – свидетельствуют об очевидном преимуществе модифицированного алгоритма.

Рассмотрим не менее интересный пример, связанный с решением задачи медицинской диагностики. Материалом для эксперимента служили архивные электрокардиограммы (ЭКГ) больных инфарктом миокарда – 44 случая и здоровых людей – 56 случаев [31]. Каждая ЭКГ характеризовалась 35 признаками, представляющими собой изменение амплитуд зубцов, их продолжительности, продолжительности некоторых сегментов и т.п. Для каждого эксперимента все 100 случаев перемешивали случайным образом и разделяли на обучающую (50) и экзаменационную (50) последовательности. Для каждого из случаев экзаменационной последовательности оценивали принадлежность его к одному из двух классов ("норма" и "патология") по приведенному выше алгоритму МЛЭ без подстройки порога. Качество алгоритма оценивалось отношением b числа неправильно классифицированных случаев ко всей экзаменационной последовательности. Для сравнения приведем усредненные по 10 различным экспериментам результаты использования трех различных локальных алгоритмов при $n = 5$:

по методу ближайшей величины $b = 0,344 \pm 0,032$;

по методу Фикса – Ходжеса $b = 0,321 \pm 0,028$;

на основе МЛЭ $b = 0,284 \pm 0,018$.

Результаты свидетельствуют о преимуществе рассмотренного алгоритма перед другими локальными процедурами.

Итак, при использовании метода многомерной линейной экстраполяции могут быть построены достаточно надежные алгоритмы решения задач распознавания образов, представляющие собой основу принятия оперативных решений как на стадии проектирования, так и на стадии управления динамической системой.

Глава III

ПРИМЕНЕНИЕ АЛГОРИТМА МНОГОМЕРНОЙ ЭКСТРАПОЛЯЦИИ ДЛЯ РЕШЕНИЯ ЗАДАЧ УПРАВЛЕНИЯ

1. МЕТОД МНОГОМЕРНОЙ ЭКСТРАПОЛЯЦИИ И ЗАДАЧИ УПРАВЛЕНИЯ ДИНАМИЧЕСКОЙ СИСТЕМОЙ

При решении многочисленных задач управления динамическими системами часто возникает необходимость оценки производных функции n -го порядка с целью повышения качества регулирования и эффективности управления. Задача эта усложняется тем, что функция, как правило, наблюдается на фоне шума различной интенсивности, причем с увеличением интенсивности шума качество восстановления производных функции (особенно производных высокого порядка) ухудшается. Не менее важный вопрос, возникающий в задачах управления динамическими системами, — умение предвидеть “будущее” состояние системы, что чаще всего приводит к необходимости решения задачи прогнозирования сигнала, поступающего на вход динамической системы.

Вплотную к названным задачам примыкает задача выбора оптимального управления по результатам наблюдения за поведением динамической системы. Эта задача особенно важна для оператора, управляющего сложной системой, который должен по результатам наблюдения фазовых координат принимать оптимальные (рациональные) решения в различных (в том числе аварийных) ситуациях.

Все перечисленные задачи могут с успехом решаться методом многомерной экстраполяции. Покажем это на примерах.

1. **Задача определения производной функции.** Для решения этой задачи была составлена программа для ЭВМ “Минск-32”, реализующая метод многомерной экстраполяции. В качестве обучающей последовательности использовались k синусоидальных линейно независимых функций и k соответствующих им l — x производных. Набор функций и их производных в некоторых точках интервала наблюдения соответствовал векторам A_j ($j = \overline{1, k}$) и X_j ($j = \overline{1, k}$). За наблюдаемую функцию принимали сигнал, похожий на обучающие синусоиды, но не совпадающий с ними. На векторах A_j строили подпространство размерности $k - 1$, определяли коэффициенты λ_j , и в соответствии с гипотезой о линейности преобразо-

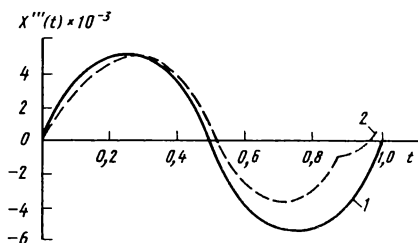


Рис. 18. Оценка третьей производной при идеальном обучении: 1 — идеальная производная; 2 — производная, полученная методом многомерной экстраполяции

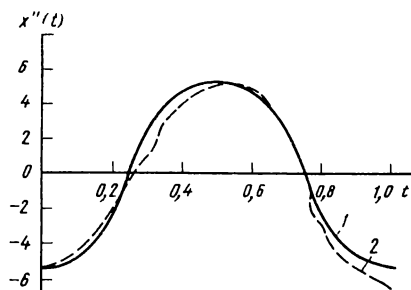


Рис. 19. Оценка второй производной при зашумленном обучении: 1 — идеальная производная; 2 — производная, полученная методом многомерной экстраполяции

вания пространства ситуаций в пространство характеристик определялся вектор X^* . Этот вектор и принимали за l -ю производную наблюдаемой функции. В целях проверки устойчивости метода к ошибкам входной информации на наблюдаемую функцию накладывали шум.

В программе формировались четыре линейно независимые обучающие функции (четыре вектора A_j) и соответствующие им l -е производные (четыре вектора X_j). Затем в программе формировался зашумленный сигнал (вектор A) той же размерности и методом многомерной экстраполяции определялась его l -я производная. Уровень шума, который накладывался на наблюдаемую функцию, составлял 20 % от амплитуды детерминированного сигнала. Закон распределения шума на указанном интервале равномерный. Результаты восстановления третьей и второй производных представлены на рис. 18, 19.

Анализ полученных результатов показывает, что метод многомерной экстраполяции может с успехом применяться для определения производной любого порядка. Причем точность оценки производной зависит не от порядка производной, а от качества обучающих последовательностей, интервала наблюдения, уровня шума, который накладывается на наблюдаемый сигнал. Из анализа результатов следует, что метод многомерной экстраполяции устойчив к ошибкам в исходной информации [22].

2. Задача прогнозирования сигнала. Задача ставилась как задача прогноза существенно нелинейной функции при условии, что на наблюдаемый сигнал наложена помеха. В качестве такой функции была выбрана синусоида. Для решения поставленной задачи был использован алгоритм многомерной экстраполяции в том виде, в каком он был сформулирован выше [22].

Обучающий "пучок" состоял из четырех близких синусоид, часть из которых представляла пространство A_j , а часть, где решалась задача прогноза, — пространство X_j . Далее формировалась наблюдаемая последовательность, длина которой менялась в зависимости от постановки задачи. На наблюдаемую функцию накладывался шум различной интенсивности. Результаты расчетов по прогнозу существенно нелинейной

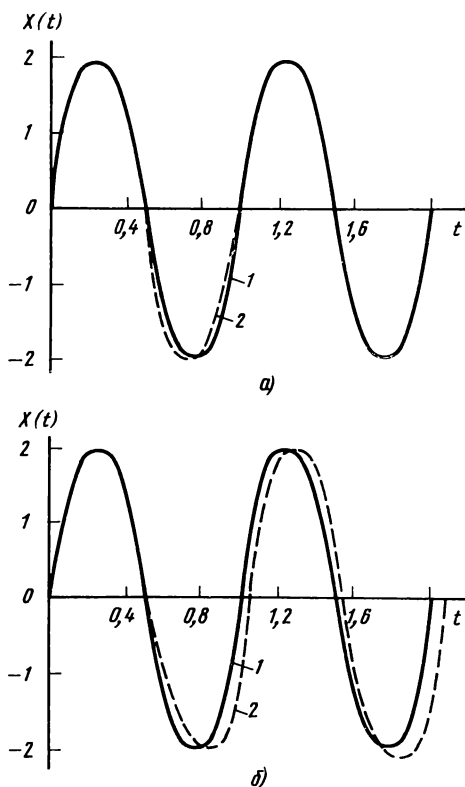


Рис. 20. Восстановление синусоиды: а — с интервалом прогноза, равным $1/2 T$; б — с интервалом прогноза, равным $1 1/2 T$: 1 — идеальная синусоида; 2 — участок синусоиды, восстановленный с помощью метода многомерной экстраполяции

функции на различные временные интервалы представлены на рис. 20. Указанные зависимости построены для постоянного интервала наблюдения, что позволяет проследить влияние длины интервала прогноза на качество прогноза.

Анализ результатов, представленных на рис. 20, а и б, показывает, что алгоритм многомерной экстраполяции может использоваться для прогноза существенно нелинейных функций. Причем рассматриваемый алгоритм позволяет решать задачу прогноза и восстановления функции

в том числе, когда интервал прогноза в несколько раз превосходит интервал наблюдения. Точность восстановления функции методом многомерной экстраполяции определяется не только объемом исходной информации и длиной интервала прогноза, но и качеством обучения (плотностью обучающих последовательностей). Этот вывод определяет преимущества метода многомерной экстраполяции, главным из которых является возможность восстановления функции по минимальной априорной информации.

3. Оценка характеристик движущегося объекта. Рассматривалась задача определения характеристик движущегося объекта по известным фазовым координатам, наблюдаемым на фоне помех. В качестве характеристик выбирали управления, воздействующие на объект.

Пусть известны фазовые координаты некоторого объекта в течение определенного интервала наблюдения. Объект этот описывается с помощью системы уравнений

$$\dot{X} = f(t, X, U),$$

где t — время; X — вектор фазовых координат; U — вектор управлений.

Требуется найти значения управлений, под воздействием которых данный объект движется по наблюдаемой траектории. Иначе говоря, имеется обучающая последовательность, которая каждой из траекторий,

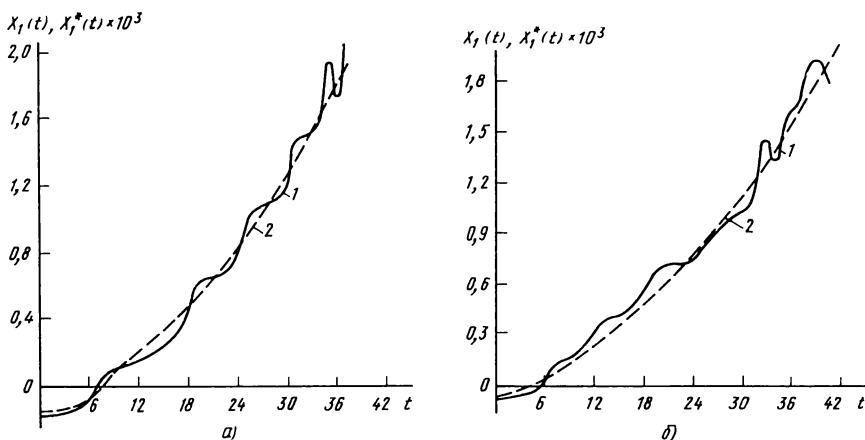


Рис. 21. Фазовые траектории: а – первый вариант формирования; б – второй вариант; 1 – наблюдаемая траектория; 2 – траектория, полученная методом многомерной экстраполяции

определяемых вектором $X_j(t)$, ставит в соответствие вектор управления $U_j(t)$. Необходимо, используя данную информацию, найти оценку вектора $U(t)$ для новой траектории, заданной вектором $X(t)$.

Для решения поставленной задачи была составлена программа для ЭВМ "Минск-32", реализующая алгоритм многомерной экстраполяции. В этой программе объект описывается с помощью следующей системы уравнений [19]:

$$\begin{aligned}\dot{X}_1 &= X_3; \\ \dot{X}_2 &= X_4; \\ \dot{X}_3 &= \frac{C}{X_5} U_1 U_2; \\ \dot{X}_4 &= \frac{C}{X_5} U_2 U_3 - g; \\ \dot{X}_5 &= U_3,\end{aligned}$$

где X_1, X_2 – декартовы координаты; X_3, X_4 – составляющие скорости объекта; X_5 – масса объекта; U_1, U_2 – направляющие косинусы вектора тяги (управления); U_3 – скорость расхода массы (управления); C – эффективная скорость истечения; g – ускорение силы тяжести.

Было просчитано два варианта. В первом варианте обучающие наборы управлений U_{1j}, U_{2j} формировали с помощью датчика случайных чисел, а управление U_{3j} принимали постоянным ($U_{3j} = \text{const}$). Соответствующие этому набору фазовые траектории $X_{1j}, X_{2j}, X_{3j}, X_{4j}, X_{5j}$ получались после интегрирования системы дифференциальных уравнений, описывающих движение объекта. На одну из траекторий накладывался шум, и ее принимали за наблюдаемую. Методом многомерной экстраполяции определяли управление, под действием которого в фазовом пространстве формировалась наблюдаемая траектория (рис. 21, а).

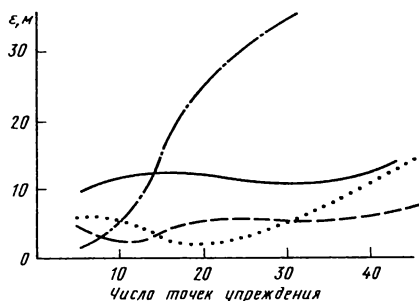
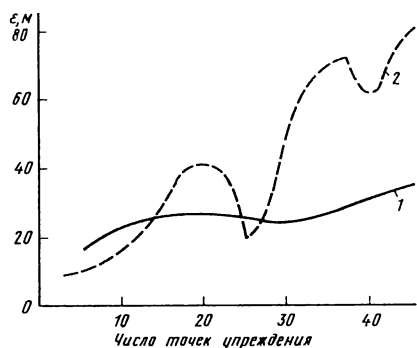


Рис. 22. Зависимость ошибки прогноза от значения интервала упреждения: 1 — по "устойчивым" формулам; 2 — по формулам Гревия

Рис. 23. Зависимость ошибки прогноза от значения интервала упреждения при различной длине обучающих последовательностей: . . . — четыре точки в обучении; — — — — — пять точек в обучении; — — — — — 10 точек в обучении; — — — — — 15 точек в обучении

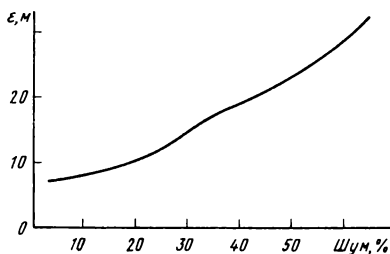
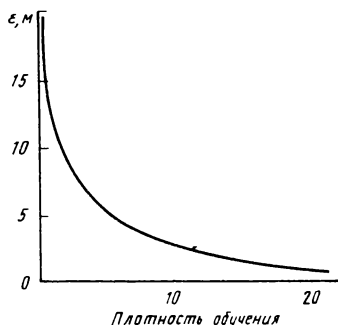


Рис. 24. Зависимость ошибки прогноза от плотности обучения

Рис. 25. Зависимость ошибки прогноза от значения шума, накладываемого на траекторию

Во втором варианте (рис. 21, б) алгоритм формирования обучающих последовательностей оставался прежним, а наблюдаемая траектория формировалась постоянными значениями управления на всем интервале наблюдений.

Анализ траекторий, представленных на рис. 21, а и б, позволяет сделать вывод о том, что метод многомерной экстраполяции является надежным инструментом прогноза управлений для случая, если движущийся объект задан системой дифференциальных уравнений.

Проведем оценку точности восстановления траектории движущегося объекта в зависимости от ряда факторов: интервала прогнозирования, длины и плотности обучающих последовательностей и т.д. Для получения количественных оценок используем следующие формулы:

$$\epsilon_1 = \frac{\sqrt{\sum_{i=k}^n (X_{i*} - X_i)^2}}{n - k};$$

$$\epsilon_2 = \frac{\sqrt{\sum_{i=k}^n (X_{2*} - X_2)^2}}{n - k};$$

$$\epsilon = \sqrt{\epsilon_1^2 + \epsilon_2^2},$$

где $n, - k$ — интервал прогнозирования; X_1^*, X_2^* — траектории, полученные методом многомерной экстраполяции; X_1, X_2 — наблюдаемые траектории; ϵ_1, ϵ_2 — ошибки по каждой координате; ϵ — ошибка прогнозирования траектории.

Результаты оценки представлены на рис. 22 — 25.

В гл. I было показано, что при использовании метода многомерной линейной экстраполяции решается вспомогательная задача выбора на гиперплоскости вектора, ближайшего к заданному. Эта задача сводится к определению псевдообратной матрицы, методы вычисления которой (формулы Гревилля) неустойчивы к возмущениям. Модификация этих формул [20] позволяет существенно улучшить результаты.

Анализ зависимостей, представленный на рис. 22, показывает, что "устойчивые" формулы [20] значительно улучшили результаты, и в дальнейших исследованиях использовались именно эти формулы.

Анализ зависимости, представленной на рис. 23, показывает, что с увеличением длины обучающих последовательностей качество прогноза улучшается. Эта закономерность наблюдается до тех пор, пока длина обучающих последовательностей не достигает порогового значения, начиная с которого наблюдается прямая зависимость (с увеличением длины обучающих последовательностей повышается ошибка прогноза). Указанная зависимость объясняется влиянием "старой" информации на качество прогноза. Отсюда следует, что алгоритм экстраполяции должен быть построен с учетом "отбрасывания" устаревшей информации, что в свою очередь требует решения задачи выбора оптимальной (рациональной) длины обучающей последовательности.

Анализ зависимости, представленной на рис. 24, показывает, что при увеличении плотности обучений до 10 ошибка прогноза уменьшается, причем довольно резко; при дальнейшем увеличении плотности обучений ошибка прогноза изменяется мало и стремится к постоянному значению. Отсюда следует, что для задач, подобных рассматриваемой в данном параграфе, плотность обучений должна быть не больше 4 — 6. Этот вывод важен с точки зрения реализации метода многомерной экстраполяции, поскольку именно объем обучающей информации определяет необходимую память ЭВМ.

Анализ зависимости, представленной на рис. 25, показывает, что хотя с увеличением значения шума ошибка прогноза растет, метод многомерной экстраполяции оказывается работоспособным при значительных ошибках измерений. Даже при значении шума, достигающем 70 % от амплитуды наблюдаемого сигнала, ошибки прогноза не превосходят реально существующих значений, необходимых для целей управления движущимися объектами.

Таким образом, из анализа экспериментальных материалов следует, что метод многомерной экстраполяции может использоваться для прог-

ноза как детерминированных, так и стохастических сигналов. Причем при прогнозе последних метод оказывается малочувствительным к уровню шума, накладываемого на наблюдаемую траекторию, что важно при решении задач управления и прогноза в реально существующих ситуациях. Обеспечение высокой точности и устойчивости метода многомерной экстраполяции предполагает решение задач, связанных с обоснованием длины обучающих последовательностей, плотности обучения, значения интервала упреждения и т.д. Многие из задач, рассмотренных выше, относятся к классу некорректных задач. Одно из преимуществ метода многомерной экстраполяции — возможность использования его для решения подобных задач. Кроме того, метод многомерной экстраполяции позволяет эффективно решать задачи оперативного управления при отсутствии модели управляемого объекта, что особенно важно для адаптации вычислительных систем. Так, алгоритмы решения задач оценки производных и экстраполяции динамики прямо могут быть использованы для адаптации эволюционирующей вычислительной системы, свойства и характеристики которой изменяются определенным, но неизвестным образом.

2. ЭКСТРАПОЛЯЦИОННЫЕ МОДЕЛИ ОПЕРАТИВНОГО УПРАВЛЕНИЯ РЕЖИМАМИ ЭЛЕКТРОЭНЕРГЕТИЧЕСКИХ СИСТЕМ

Кардинальное повышение надежности и экономичности функционирования объединенных электроэнергетических систем (ОЭС) в настоящее время связано с созданием и внедрением в практику эксплуатации автоматизированных систем диспетчерского управления (АСДУ) режимами работы ОЭС [40]. С помощью АСДУ можно осуществлять сбор, обработку и отображение информации о текущих режимах ОЭС, оперативное формирование оптимальных решений, по управлению режимами ОЭС в аварийных и незапланированных нормальных ситуациях.

Во многих случаях получение тех или иных оптимальных управляющих воздействий при помощи точных численных математических моделей требует несоизмеримо больших затрат машинного времени по сравнению с требуемым временем принятия решений диспетчерским персоналом либо подсистемой АСДУ. Например, оптимальная корректировка распределения нагрузок между электростанциями современных ОЭС может потребовать затрат машинного времени порядка 30 — 90 мин [40]. Расчет динамической и статической устойчивости современных ОЭС даже на ЭВМ ЕС-1030 требует затрат машинного времени порядка 10 — 60 мин.

В этих условиях для целей эффективного оперативного управления необходимо в рамках АСДУ иметь модели, отражающие стохастические (вследствие стохастической природы режима ОЭС) связи между показателями функционирования ОЭС (зависимыми переменными) и признаками (независимыми переменными), определяющими интересующие нас режимные ситуации.

Под режимной ситуацией ω подразумевается состояние режима

работы ОЭС, информация о параметрах которого с той или иной полнотой может быть получена (измерена) в любой момент времени. Обозначим множество всех возможных ситуаций Ω , тогда $\omega \in \Omega$. Выборка $W = \{\omega_1, \omega_2, \dots, \omega_N\}$ — совокупность из N ситуаций, используемых при синтезе модели ($W \in \Omega$). Каждая режимная ситуация ω характеризуется парой векторов $\{X, Y\}$, представляющих собой совокупность из n независимых переменных $X^T = (x_1, \dots, x_n)$ и m зависимых от них переменных $Y^T = (y_1, \dots, y_m)$. Совокупность значений независимых переменных режимной ситуации составляет ее описание. Множество всех возможных описаний образует евклидово пространство E^n размерности n , т.е. $X \in E^n$. Совокупность значений зависимых переменных режимной ситуации составляет ее решение. Множество всех возможных решений образует евклидово пространство E^m размерности m , т.е. $Y \in E^m$. Любая режимная ситуация ω_i отображается в точки X^i и Y^i пространств E^n и E^m .

В общем случае синтез модели состоит в восстановлении с тем или иным приближением неизвестного стохастического оператора отображения $X \in E^n$ в $Y \in E^m$ на основании выборочной совокупности $\{X^i, Y^i\}_{i=1}^N$

$$Y = \Psi(X).$$

Для целей оперативного управления режимами ОЭС оператор отображения $\Psi(X)$ во многих случаях целесообразно представлять в виде приближенной функциональной зависимости

$$Y = \Phi_c(B_c, X) + \Delta,$$

где Δ — ошибка; c — индекс, указывающий, в каком классе решается задача восстановления функции; B_c — вектор параметров в выбранном классе функций Φ_c .

Таким образом, возникает задача восстановления функциональной зависимости $\Phi_c(B_c, X)$. Ошибка Δ появляется, во-первых, вследствие приближенного соответствия между стохастическим отображением $Y = \Psi(X)$ и выбранной аппроксимирующей функцией $\Phi_c(B_c, X)$; во-вторых, из-за неучета ненаблюдаемых независимых переменных; в-третьих, вследствие ограниченного объема выборки, на основании которой восстанавливается функция $\Phi_c(B_c, X)$.

Восстановление функции $\Phi_c(B_c, X)$ принципиально может быть осуществлено на основе всей либо части выборки W . В первом случае получаем кусочно-непрерывную модель (в частном случае непрерывную) с однородной структурой [40], во втором — модель с неоднородной структурой, зависящей от значений вектора описаний X . Гиперповерхности, описывающие связь между переменными большинства оперативных моделей управления режимами ОЭС, существенно нелинейны в реальной области изменения переменных, структура их непостоянна во времени и в значительной степени зависит от топологии сети и состава оборудования. Вследствие этого модели с однородной структурой обеспечивают, как правило, недостаточную для практики точность решения Y , требуют частого пересчета параметров моделей.

В большинстве случаев разумно предположить, что близким в соответствии с выбранной мерой близости описаниям режимных ситуаций X в

пространстве E^n соответствуют близкие решения Y в пространстве E^m . В этой связи, очевидно, трудоемкая процедура восстановления функции $\Phi_c(B_c, X)$ во всей области изменения X может быть заменена построением локальной модели $\Phi_c(B_c, X^t)$, отражающей функционирование ОЭС лишь в некоторой окрестности ее текущего состояния X^t , для которого необходимо найти решение Y^t . Преимущество такого подхода — простота выбора класса локальной функции $\tilde{\Phi}_c(B_c, X)$. В частности, можно использовать линейные или даже константные функции. Для их восстановления в окрестности X^t необходимо знать лишь небольшое число пар $\{X^i, Y^i\}_{i=1}^L$, $L \leq N$ из этой окрестности.

Локальные линейные модели строят, например, с использованием линейных членов разложения в ряд Тейлора, при помощи коллектива линейных решающих правил, кусочно-линейной и нелинейной аппроксимации, различных методов интерполяции и другими методами. Однако при реализации всех этих методов число используемых пар $\{X^i, Y^i\}_{i=1}^L$ должно быть больше размерности пространства независимых переменных ($L > n$). При использовании же моделей на практике число пар выборки $\{X^i, Y^i\}_{i=1}^L$ в окрестности текущей режимной ситуации X^t , как правило, оказывается меньше n ($L < n$). И в этом случае с успехом могут использоваться локальные линейные модели, основанные на методе многомерной экстраполяции. Экстраполяционные модели относятся к классу адаптивных обучающихся моделей, поскольку решающие правила для получения оценок \hat{Y}^t на основе метода МЛЭ зависят от X^t .

Как уже было показано в гл. I, точность оценки Y^t , определяемая значением абсолютной ошибки $\Delta = |Y^t - \hat{Y}^t|$, зависит от числа выбранных пар (X, Y) , расположения $\{X^i\}_{i=1}^L$ в пространстве E^n и близости этих пар к X^t (при заданной метрике ρ). Увеличение числа таких пар l , используемых для экстраполяции, от двух до $n + 1$ не означает монотонного снижения ошибки Δ .

В этой связи возникает задача выбора оптимального решающего правила — комбинации пар $\{X^i, Y^i\}_{i=1}^L$, $L = \overline{2, N}$ из всех N пар выборки, обеспечивающей максимально возможную точность оценки Y^t . Число используемых пар l будет определять класс решающих правил; тогда конкретное правило класса будет обозначаться S_ν^L , где ν — номер правила из класса L . В силу локальности алгоритма МЛЭ оптимальное решающее правило необходимо строить только из пар ($L < N$), принадлежащих ближайшей окрестности в смысле метрики ρ . Окрестность из L пар может задаваться разными способами, например, фиксированным числом L ближайших к X^t точек, радиусом r с центром в X^t , который также выделяет L ближайших точек. Пары $\{X^i, Y^i\}_{i=1}^L$ и определяют локальную выборку X^t . Однако число пар l , необходимых для построения оптимального решающего правила, зависит от неизвестной функции $\tilde{\Phi}_c^t(B_c, X^t)$, а также размерности вектора X^t . Можно показать, что при линейном характере $\tilde{\Phi}_c^t(B_c, X^t)$ в окрестности X^t и попадании в эту окрестность $L > n$ пар точная оценка Y^t ($\Delta = 0$) обеспечивается при $l = n + 1$. При других функциях ничего нельзя сказать о числе необходимых пар в оптимальном решающем правиле. Общее число решающих правил

Рис. 26. Геометрическая интерпретация проблемы выбора решающего правила

H_s в окрестности X^t определяется числом всех возможных комбинаций из L по l , т.е.

$$H_s = \sum_{l=2}^L C_L^l.$$

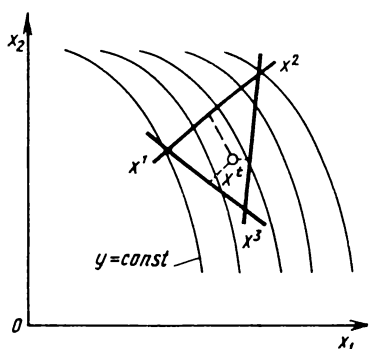
Рассмотрим геометрическую интерпретацию проблемы выбора оптимального решающего правила для одномерного вектора зависимых параметров y ($m = 1$). Пусть вектор параметров описания X двумерный ($n = 2$). Рассмотрим пространство векторов $X \in E^2$ (рис. 26), где показаны линии равного уровня $y = \text{const}$ и три точки X^i $i=1, 3$, для которых известны значения функции. Эти точки, как видно на рисунке, попали в практически линейную область.

С целью иллюстрации сравнения различных решающих правил рассмотрим только правила с $l = 2$, т.е. правила S^2 , использующие две пары $\{x^i, y^i\}$ $i=1, 2$. В этом случае проекция точки x^t на прямую (в общем случае при $l > 2$ — гиперплоскость), проходящую через две любые точки x , определяет \hat{x}^t и, соответственно, \hat{y}^t . Общее число возможных решающих правил $H_s^2 = C_3^2 = 3$. Как видно на рис. 26, наилучшим решающим правилом для этой области будет правило, построенное из точек x^1 и x^2 , т.е. прямая, приближающаяся к ортогональной относительно линий равного уровня.

Таким образом, если выделенная окрестность вектора X^t близка к линейной и число пар L , попавших в эту окрестность, меньше $n + 1$, то целесообразно решать задачу выбора наилучшего для данной окрестности решающего правила. Предварительной задачей при этом является выделение локальной окрестности вектора X^t в E^n . Пары, попавшие в эту окрестность, могут быть использованы для формирования локальной обучающей выборки и конструирования решающих правил S_v^l , $l = \overline{2, L}$. Синтез оптимальных решающих правил может осуществляться на основе минимизации эмпирического риска [5].

Рассмотренная выше процедура получения решения \hat{y}^t для каждой новой ситуации X^t требует трудоемких вычислительных операций, связанных с выделением локальной окрестности X^t , и обучения оптимальному решающему правилу. При решении задачи оперативного управления режимами ОЭС можно использовать более экономичный подход. Вместо задачи выделения локальной окрестности для каждого нового X^t , подлежащего оценке, решается предварительная задача разбиения всего пространства X (всей обучающей последовательности) на области (группы), имеющие общее оптимальное решающее правило. Эту задачу можно решать в два этапа.

Первый заключается в разбиении всего пространства режимных ситуаций X выборки W (выборочного пространства) на однородные области, исходя из физических соображений и иерархичности структуры



управления ОЭС. Такие области можно отождествлять, например, с характерными периодами режима ОЭС в течение суток: ночной и дневной минимум, утренний и вечерний максимум. При разбиении выборочного пространства следует исходить из требования линейности функции $\Phi_c(B_c, X)$ в каждой из областей.

Второй этап заключается в последующем дроблении полученных областей на подобласти методами таксономии [11].

При обсуждении алгоритмов синтеза экстраполяционных моделей предполагалось, что описание X и набор искомых решений Y заданы. Выбор Y не представляет затруднений, поскольку зависит от известной цели использования модели. Сложнее обстоит дело с выбором описания.

На начальном этапе конструирования модели, как правило, не имеется достаточных априорных сведений о том, какая именно совокупность независимых переменных существенно влияет на рассматриваемые показатели функционирования ОЭС. Задача выбора переменных на этом этапе решается методами экспертных оценок. Как правило, выбранное описание избыточно; некоторые из введенных в рассмотрение признаков либо находятся в тесной корреляционной связи, либо практически не влияют на зависимые переменные. Таким образом, возникает задача сокращения размерности признакового пространства n . Решение данной задачи позволяет:

- отобрать существенные (статистически значимые, информативные) признаки;

- повысить достоверность принимаемых решений при помощи экстраполяционных моделей;

- выявить тесную связь между некоторыми признаками;

- упростить использование модели в управлении режимами ОЭС;

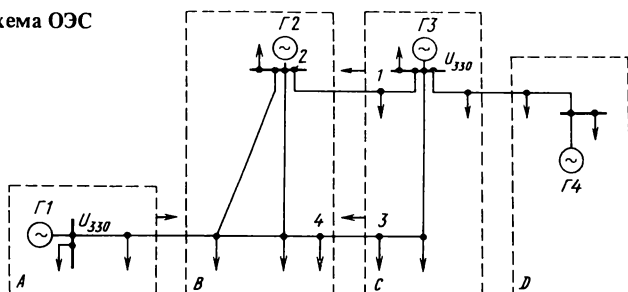
- уменьшить затраты на получение выборочной информации и пересчет параметров модели с течением времени.

Можно выделить три группы методов сокращения признакового пространства. К первой группе относятся методы сокращения описания с одновременным преобразованием признакового пространства — факторный анализ, метод главных компонент. Данные методы параметрические и для достоверной оценки результатов требуют, чтобы соблюдалась гипотеза нормальности и неизменности параметров совместного закона распределения независимых переменных. К сожалению, для многих стохастических моделей функционирования ОЭС подобная гипотеза ее выполняется, что ограничивает применимость подобных методов.

Ко второй группе относятся методы классификации, выявляющие в пространстве E^n группы признаков, тесно связанных между собой в соответствии с выбранной мерой близости. Из каждой группы выбирается типичный представитель, и из подобных представителей составляется признаковое пространство меньшей размерности.

Перечисленные методы, как ясно из изложенного, используются до процесса восстановления функции $\Phi_c(B_c, X)$. Они не учитывают в явном виде структуру экстраполяционных моделей, т.е. не зависят от цели исследования. Поэтому выбор значимых признаков при помощи данных методов может привести к потере информации об изучаемых

Рис. 27. Обобщенная схема ОЭС



показателях функционирования ОЭС, а значит, и к ухудшению точности экстраполяционных моделей режимных ситуаций.

Наиболее конструктивным представляется третий путь оптимального выбора описания применительно к экстраполяционным моделям — сочетание методов группировки признаков и оценки значимости признаков по их влиянию на точность экстраполяционных моделей режимных ситуаций, т.е. на основе минимизации эмпирического риска. Подобная двухуровневая процедура позволяет исключить эффект дублирования признаков уже до синтеза модели. В процессе же синтеза экстраполяционной модели режимных ситуаций выявляются существенные независимые переменные.

Рассмотрим в качестве примера экстраполяционную модель предельной пропускной способности межсистемных ЛЭП по условиям сохранения статической аperiodической устойчивости применительно к одной из реально функционирующих ОЭС [40].

На обобщенной схеме ОЭС (рис. 27), состоящей из четырех систем (A, B, C, D), стрелками указаны направления потоков активной мощности по межсистемным связям в нормальных режимах. Систематизирующим является напряжение 330 кВ. Задачу решали с заданным числом независимых переменных, определяемых экспериментальным путем. В качестве независимых переменных были выбраны следующие признаки, информацией о которых располагает диспетчерский персонал:

- напряжение x_1 на шинах 330 кВ генератора ГЗ, кВ;
- напряжение x_2 на шинах 330 кВ генератора Г1, кВ;
- суммарное электропотребление x_3 энергосистемы В, МВт;
- загрузка x_4 по активной мощности Г2, МВт;
- переходная ЭДС x_5 Г2, кВ;
- эквивалентное реактивное сопротивление x_6 Г2, Ом.

В качестве зависимой переменной y рассматривалось значение пропускной способности между энергосистемами С и В (ЛЭП 1 — 2 и 3 — 4) в направлении от С к В.

Таким образом, описанная режимная ситуация характеризовалась шестимерным вектором X в евклидовом пространстве E^6 . Все шесть независимых переменных нормировались в интервале 0 — 1.

Выборка представляла собой $N = 100$ режимных ситуаций. С целью эффективного использования выборки эксперимент проводился методом скользящего контроля, т.е. последовательно выбиралась каждая из 100 режимных ситуаций и с помощью оставшихся 99 пар $\{x^i, y^i\}_{i=1}^{99}$, которые служили обучающей последовательностью, оценивалась величина \hat{y}^i .

Реализованы следующие алгоритмы:

экстраполяция "по l ближайшим" к X^t ситуациям;

выделение локальной окрестности заданием радиуса r с центром в X^t и последующим выбором в ней наилучшего решающего правила путем минимизации эмпирического риска.

В алгоритме "по l ближайшим ситуациям" $l = 2, 3, 4, 5, 6$. Ниже приведены значения (в процентах) средних значений $\bar{\delta}$ и средних квадратических отклонений σ_{δ} относительных ошибок $\delta = (y^t - \hat{y}^t)/y^t$ оценок y^t .

l	2	3	4	5	6
$\bar{\delta}$	-0,50	-0,26	0,32	-0,27	0,24
σ_{δ}	5,21	4,63	3,28	3,97	10,01

Для второго типа алгоритма радиус $r = 0,02; 0,04; 0,06; 0,08; 0,10; 0,12; 0,14$.

Ниже приведены значения $\bar{\delta}$ и σ_{δ} для всех значений радиуса.

r	0,02	0,04	0,06	0,08	0,10	0,12	0,14
$\bar{\delta}$	0,12	-0,44	0,18	-0,24	-0,41	-0,16	-0,63
σ_{δ}	5,46	3,86	3,42	3,38	3,24	3,91	4,82

Анализ приведенных результатов показывает, что средние значения ошибок близки к нулю. Поэтому в качестве меры точности решающих правил целесообразно использовать средние квадратические отклонения ошибок. Для указанных алгоритмов зависимости $\sigma_{\delta} = U(l)$ и $\sigma_{\delta} = U'(r)$ имеют экстремумы соответственно при $l = 4$ и $r = 0,1$. Наличие экстремумов объясняется тем, что гиперповерхность предельных значений пропускной способности ЛЭП $Y = \Psi(X)$, как показали исследования (на основе синтеза регрессионных моделей), существенно нелинейная. В этом случае увеличение числа точек, используемых в обучающей выборке в обоих алгоритмах приводит к снижению σ_{δ} до тех пор, пока $Y = \Psi(X)$ в пределах изменения независимых переменных обучающей выборки практически линейна. Следовательно, для определения предела статической апериодической устойчивости в новой режимной ситуации наиболее рационально строить решающие правила следующим образом:

в алгоритме "по l ближайшим точкам" следует выбрать четыре точки ($l = 4$);

в алгоритме с заданным радиусом следует минимизировать эмпирический риск в радиусе $r = 0,1$.

Результаты обучения показывают, что оба алгоритма обеспечивают приблизительно одинаковую точность, вполне приемлемую для практики.

В заключение следует отметить, что при прогнозировании пропускной способности ЛЭП в послеаварийных ситуациях, когда регламентируемый запас устойчивости не превышает 8 %, целесообразно выбирать такие решающие правила, которые минимизируют только положительные ошибки. В этом случае, очевидно, увеличивается надежность параллельной работы энергосистемы.

3. СИНТЕЗ АЛГОРИТМОВ ПОИСКОВОЙ ОПТИМИЗАЦИИ МЕТОДОМ МЛЭ

Рассмотрим задачу безусловной минимизации скалярной функции $Q(X)$ в n -мерном евклидовом пространстве E^n :

$$Q(X) \rightarrow \min_{X \in E^n} \Rightarrow X^*,$$

где X^* — решение этой задачи, т.е. $Q(X^*) = \min Q(X)$.

Известны различные итеративные процедуры поиска локального минимума X^* , например, метод Ньютона, покоординатного спуска, градиентные методы и т.д. Эти методы, по сути дела, универсальны, так как почти не учитывают специфики класса решаемых задач. Кроме того, как правило, данные методы не гарантируют сходимости к решению X^* из произвольных начальных точек. Так, например, метод Ньютона нельзя применять при наличии седловых ситуаций.

Для заданного класса задач, т.е. для заданного множества $\{Q(X)\}$, всегда можно синтезировать метод, соответствующий именно этому классу, ориентированный только на этот класс. Рассмотрим задачу синтеза такого алгоритма оптимизации, если класс минимизируемых функций $\{Q(X)\}$ задан каким-либо образом. Пусть I — информация об этом классе. Тогда одношаговый алгоритм оптимизации можно записать в виде рекуррентной формулы

$$X_{l+1} = X_l + F(X_l, I_l),$$

где $I_l \in I$ — информация о классе, используемая на l -м шаге, а F — функция, определяющая алгоритм поиска.

Как видно, эта функция и задает алгоритм. Поэтому, выбирая то или иное представление функции F и информации о классе I , можно построить различные одношаговые алгоритмы поиска.

Для синтеза и представления функции F воспользуемся методом многомерной экстраполяции [8]. Синтез функции F является в то же время синтезом алгоритма поиска оптимума X^* . Естественно, что наибольший интерес представляет синтез оптимального алгоритма, приводящего из любой исходной точки X к оптимуму X^* , т.е. за одну итерацию

$$X^* = X + F(X, I_X),$$

где I_X — информация о классе, используемая в точке X .

Отсюда получаем выражение для оптимального приращения аргумента в произвольной исходной точке X

$$\Delta X^* = X^* - X = F(X, I_X).$$

Синтез алгоритма оптимизации предполагает соответствующее представление для I_X . Эта информация должна отражать опыт оптимизации класса в районе точки X .

Пусть произвольная точка X характеризуется некоторым вектором ситуации A , компонентами которого служат, например, значение функции $Q(X)$ в этой точке, значения ее производных и т.д. Информация о классе решаемых задач представляется в виде выборки, определяющей соответ-

ствия между векторами ситуаций и оптимальным решением в этих ситуациях:

$$I: A_i \rightarrow \Delta X_i^*, i = \overline{1, L}.$$

При поиске решения в произвольной "промежуточной" точке X ситуация A , ее характеризующая, экстраполируется по k ближайшим ситуациям обучающей выборки I . Здесь должно выполняться условие

$$1 \leq k \leq m + 1,$$

где m – размерность пространства ситуаций $\{A\}$.

В соответствии с идеей метода многомерной линейной экстраполяции приближенное решение ΔX^* для произвольного аргумента X , характеризованного ситуацией A , можно найти в результате одной экстраполяционной процедуры, т.е.

$$\Delta X^* = \sum_{i=1}^k \lambda_i \Delta X_i^*. \quad (98)$$

Заметим, что в случае $k = 1$, когда необходимо принимать решение в ситуации A , зная решение ΔX_1^* только в ближайшей ситуации A_1 , решение ΔX^* совпадает с ΔX_1^* . Это означает, что в ситуации A принимается такое же решение, как и в ближайшей ситуации из I .

Исследуем работоспособность изложенного подхода для синтеза алгоритмов поисковой оптимизации на ряде модельных задач. Класс минимизируемых функций будем задавать параметрически, т.е. вводя функцию $Q(X, P)$, где P – вектор параметров, определяющих конкретную задачу оптимизации. Выбором параметров этих функций из определенной области и задается реализация класса. Далее при случайно выбранных значениях X измеряют параметры функции, т.е. строят вектор ситуации A , и, зная экстремум X^* этой функции, определяют оптимальное решение

$$X: \Delta X^* = X^* - X.$$

Пример 1. Рассмотрим задачу определения вещественного корня кубической параболы

$$y = (x + b)^3 + c. \quad (99)$$

В этом случае класс определяется видом функции $y = y(x, P)$, где $P = (b, c)$ – параметры класса.

Известно, что в одномерном пространстве минимизацию непрерывной и дифференцируемой функции можно заменить определением корня ее производной. Поэтому поиск корня кубической параболы, по сути дела, является минимизацией полинома четвертой степени.

Корень кубической параболы вида (99) определяется как

$$x^* = -b - \sqrt[3]{c},$$

а смещение из произвольной точки x имеет вид

$$\Delta x^* = x^* - x = -\sqrt[3]{c} - (x + b).$$

Это смещение приводит к экстремальной точке x^* . Однако здесь неизвестны параметры b и c , значения которых можно восстановить по значениям функции $y(x)$ и ее производной $y'(x)$ в точке x . Решение вычисляется по формуле, которую легко получить из формулы (99) и выражения производной $y'(x) = 3(x + b)^2$:

y	-1	-0,75	-0,5	-0,25	0,25	0,5	0,75	1
y'	2,989	2,464	1,875	1,172	1,209	1,904	2,489	3,012
Δx^*	1,177	1,085	0,970	0,804	-0,456	-0,618	-0,732	-0,823
Δx	1,077	1,085	0,970	0,704	-0,856	-0,518	-0,732	-1,023

$$\Delta x^* = \sqrt[3]{(y'/3)^3 - y} - \sqrt{y'/3} \quad (100)$$

В роли обучающей выборки естественно брать соответствия между решением Δx^* и ситуацией, определяемой y и y' . Имеем, таким образом, двумерное пространство ситуаций и одномерное пространство решений. В данном примере обучающая выборка состояла из восьми соответствий вида $(y, y') \rightarrow \Delta x^*$. Три верхние строки табл. 10 показывают обучающую выборку для конкретной кривой при $b = -0,028$ и $c = -0,006$, имеющей корень в точке $x^* = 0,207$.

Чтобы оценить точность экстраполяции по этой выборке, был проведен следующий эксперимент. Для случайно выбранного x из интервала $[-1; 1]$ вычисляли вектор ситуации (y, y') , экстраполицией по двум ближайшим точкам обучающей выборки оценивали решение Δx и определяли ошибку

$$e = |\Delta x^* - \Delta x|.$$

Средняя ошибка оценки корня на основе 100 экспериментов оказалась $\bar{e} = 0,127$.

С целью уменьшения полученной ошибки обучающую выборку подстраивали методом случайного поиска с возвратом [32]. На каждом шаге решения Δx случайно изменяли на величину $\pm 0,05$. Для полученной таким образом обучающей выборки на основе 100 экспериментов определяли среднюю ошибку вычисления корня. В результате 300 шагов поиска была получена средняя ошибка $\bar{e} = 0,107$. Четвертая строка табл. 10 показывает решения, полученные в результате подстройки.

Пример 2. Покажем, что метод многомерной линейной экстраполяции в задачах рассмотренного типа успешно конкурирует со стандартными регрессионными методами. Для этого сравним точность решения, получаемого по экстраполяционной и регрессионной моделям, на основе обучающей выборки, состоящей из 100 случайных пар вида $(y, y') \rightarrow \Delta x^*$. Как и в предыдущем примере, объектом исследования является кубическая парабола, имеющая решение (100). Точки P_i , $i = 1, 8$ соответствуют ситуациям обучающей выборки, полученным для кривой $y = (x - 1)^3 + 1$ с корнем в точке P_0 при следующих значениях x : -1,0; -0,5; 0,5; 1,0; 1,5; 2,0; 2,5; 3,0.

Для векторов ситуаций (y, y') значения взяты случайно из интервала $[0; 2]$, а значения y' из интервала $[0,01; 6]$. Экстраполяция проводилась по двум ближайшим точкам построения таким образом обучающей выборки. Средняя ошибка вычисления корня, полученная на основе 200 экспериментов, $\bar{e} = 0,476$.

На той же обучающей выборке построены три регрессионные модели – линейная, квадратичная и кубическая:

$$\Delta x = 0,158 y;$$

$$\Delta x = 0,637 y - 0,961 y^2 + 0,319 y y';$$

$$\Delta x = 2,422 y - 4,614 y^2 + 0,483 y y' + 1,712 y^3 + 0,059 y (y')^2 - 0,342 y^2 y'.$$

С помощью тех же 200 случайных векторов (y, y') , которые использовались в экстраполяционной модели, получены следующие средние ошибки для регрессионных моделей:

$$\bar{e} = 0,975 \text{ – для линейной;}$$

$$\bar{e} = 1,189 \text{ – для квадратичной;}$$

Таблица 11

y_1	-1	-0,75	-0,5	-0,25	0,25	0,5	0,75	1
y_2	-0,501	-0,344	-0,196	0,063	0,546	0,961	1,344	1,711
Δx^*	0,693	0,604	0,493	0,338	-0,923	-1,098	-1,218	-1,312
Δx	0,693	0,604	0,393	0,238	-1,023	-0,998	-1,318	-1,312

$\bar{e} = 1,339$ – для кубической.

Из анализа этих результатов следует, что учет локальных свойств пространства ситуаций при выборе ближайших точек в экстраполяционной модели позволяет получить в среднем более точное решение, чем с помощью стандартных регрессионных моделей.

Пример. 3. Обучающая выборка может не использовать производных: достаточно ограничиться значениями функций. Рассмотрим кубическую параболу $y = ax^3 + c$. Пусть в точках x_1 и x_2 известны значения этой функции $y_1 = ax_1^3 + c$; $y_2 = ax_2^3 + c$, располагая которыми можно определить смещение к корню $\Delta x^* = x^* - x_1$;

$$\Delta x^* = -x_1 + \sqrt{\frac{y_1 x_2^3 - y_2 x_1^3}{y_1 - y_2}}.$$

Обучающая выборка составлена из восьми соответствий вида $(y_1, y_2) \rightarrow \Delta x^*$, где значение y_2 вычислено в точке $x_2 = x_1 + 0,2$ (табл. 11).

Три верхние строки табл. 11 показывают исходную обучающую выборку для функции $y = 0,992 x^3 + 0,033$ с корнем в точке $x^* = -0,32$. Как и в предыдущих примерах, определяем среднюю ошибку вычисления решения.

Для исходной обучающей выборки эта ошибка была $\bar{e} = 0,112$. С целью уточнения приближенного решения обучающая выборка подстраивалась методом случайного поиска с возвратом. Результат такой подстройки на основе 300 шагов случайного поиска представлен в четвертой строке табл. 11. Средняя ошибка вычисления корня в этом случае снизилась до $\bar{e} = 0,092$.

Пример. 4. Выше рассматривался двухпараметрический класс задач определения корня. Можно исследовать и более сложную кубическую параболу вида $y = a(x + b)^3 + c$, для которой

$$\Delta x^* = x^* - x = -\sqrt[3]{c/a} - (x + b).$$

Если в точке x известны значения y, y', y'' , то решение о смещении вычисляется по формуле

$$\Delta x^* = \frac{2y'}{y''} \left(\sqrt{1 - \frac{3yy''}{2(y')^2}} - 1 \right),$$

где предполагается, что y' и y'' не равны нулю.

Для конкретной функции $y = 0,954(x - 0,012)^3 + 0,034$, имеющей корень в точке $x^* = -0,317$, обучающая выборка была составлена из восьми соответствий вида $(y, y', y'') \rightarrow \Delta x^*$ (табл. 12).

Оказалось, что если в обучающей выборке учитывается информация о первой и второй производных функции, то метод многомерной линейной экстраполяции

Таблица 12

y	-1	-0,75	-0,5	-0,25	0,25	0,5	0,75	1
y'	3,020	2,511	1,994	1,276	1,064	1,775	2,364	2,886
y''	-5,879	-5,361	-4,717	-3,821	3,489	4,508	5,202	5,748
Δx^*	0,698	0,608	0,495	0,339	-0,938	-1,116	-1,238	-1,333

y_1	-1	-0,75	-0,5	-0,25	0,25	0,5	0,75	1
y_2	-0,506	-0,347	-0,198	-0,064	0,540	0,953	1,334	1,700
y_3	-1,729	-1,367	-0,991	-0,589	0,099	0,227	0,374	0,530
Δx^*	0,698	0,608	0,495	0,339	-0,938	-1,116	-1,238	-1,333

гарантирует высокую точность решения, в результате чего не требуется настройка обучающей выборки методом случайного поиска. Средняя ошибка вычисления корня с применением экстраполяции по двум ближайшим точкам $\bar{\epsilon} = 0,005$. Экстраполяция по трем ближайшим точкам обучающей выборки позволила уменьшить значение ошибки до $\bar{\epsilon} = 0,0012$, а по четырем ближайшим точкам из произвольной точки x привела к точному значению корня x^* .

Пример 5. Оценка производных для функций, заданных таблично, вызывает неудобства. Поэтому естественно ограничиться лишь значениями самой функции. В данном примере демонстрируется возможность такого образования обучающей выборки. Исследуется та же кубическая парабола, что и в предыдущем примере. Для определения корня можно использовать три значения функции. Поэтому обучающая выборка образована соответствиями вида $(y_1, y_2, y_3) \rightarrow \Delta x^*$, где y_1, y_2, y_3 – значения функции в точках $x_1, x_2 = x_1 + 0,2, x_3 = x_1 - 0,2$ (табл. 13).

Для случайного x_1 из интервала $[-1; 1]$ вычислялся соответствующий вектор ситуаций (y_1, y_2, y_3) , и экстраполяцией по k ближайшим векторам обучающей выборки находили приближенное решение Δx . На основе 100 экспериментов определялась средняя ошибка вычисления корня. При $k = 2$ ошибка $\bar{\epsilon}$ оказалась равной 0,095, при $k = 3$ $\bar{\epsilon} = 0,062$ и, наконец, при $k = 4$ $\bar{\epsilon} = 0,0004$. Как видно, и в этом случае не требуется подстройки обучающей выборки, так как точность определения корня достаточно высока.

Выше рассмотрены различные способы заполнения, настройки и использования обучающей выборки при поиске корня кубической параболы методом многомерной линейной экстраполяции. Теперь обратимся к задачам оптимизации.

Пример. 6. При минимизации квадратичной функции $y(x) = a_0 + a_1 x + a_2 x^2$ обучающую выборку строят из соответствий вида $(y, y', y'') \rightarrow \Delta x^*$, где

$$\Delta x^* = x^* - x = -(a_1/2a_2) - x.$$

Это решение можно выразить при помощи компонент вектора ситуации

$$\Delta x^* = -y''/y''.$$

Произвольная ситуация экстраполируется по двум ближайшим ситуациям обучающей выборки. Для учета того, что при $y' = 0$ решение тоже должно быть нулевым, в правило выбора ближайших ситуаций вводится дополнение: если расстояние от произвольной ситуации (y, y', y'') до координатной плоскости, содержащей ситуацию $(y, 0, y'')$, меньше расстояния до одной из ближайших точек, то эта точка при экстраполяции изменяется ситуацией $(y, 0, y'')$. Эксперимент на ЭВМ, приведенный для квадратичной функции $y(x) = -0,176 - 0,091x - 0,021x^2$ и 50-ти соответствий выборки, показал, что в результате одной процедуры экстраполяции из произвольной точки x , принадлежащей интервалу $[-1; 1]$, приходим к практически точному решению $x^* = 2,197$.

Пример 7. Рассмотрим в двумерном пространстве минимизацию квадратичной формы

$$y(x_1, x_2) = a_0 + a_1 x_1 + a_2 x_2 + a_3 x_1^2 + a_4 x_2^2 + a_5 x_1 x_2.$$

Для точки экстремума имеем следующие выражения:

$$x^* = \frac{a_2 a_5 - 2 a_1 a_4}{\Delta}; \quad y^* = \frac{a_1 a_5 - 2 a_2 a_3}{\Delta},$$

где $\Delta = 4a_3 a_4 - a_5^2 \neq 0$. Чтобы точка (x_1^*, x_2^*) была точкой локального минимума, необходимо собственные значения характеристического уравнения

$$\begin{vmatrix} 2a_3 - \mu & a_5 \\ a_5 & 2a_4 - \mu \end{vmatrix} = 0$$

иметь положительными. Значит, коэффициенты квадратичной формы $y(x_1, x_2)$ должны удовлетворять условию

$$a_3 + a_4 > \sqrt{(a_3 - a_4)^2 + a_5^2}.$$

Для конкретной квадратичной формы

$$y(x_1, x_2) = -2 + x_1 x_5 + 5x_1^2 + 2x_2^2 + 4x_1 x_2,$$

принимавшей минимальное значение $y^* = -2,458$ в точке $(x_1^*, x_2^*) = (-0,333; 0,583)$, приводился следующий эксперимент. Измеряли значения квадратичной формы в пяти точках:

$$y_0 = y(x_1, x_2); y_1 = y(x_1 + c, x_2); y_2 = y(x_1 - c, x_2);$$

$$y_3 = y(x_1, x_2 + c); y_4 = y(x_1, x_2 - c).$$

где $c = 0,2$, а значения x_1 и x_2 случайно выбраны из интервала $[-1; 1]$. Затем для векторов ситуаций вычисляли приращения функции

$$\Delta y_1 = y_1 - y_0; \Delta y_2 = y_2 - y_0; \Delta y_3 = y_3 - y_0;$$

$$\Delta y_4 = y_4 - y_0.$$

В соответствии обучающей выборки использовали приращения аргумента

$$\Delta x_1^* = x_1^* - x_1; \Delta x_2^* = x_2^* - x_2.$$

Таким образом, для обучающей выборки были построены 50 соответствий вида

$$(\Delta y_1, \Delta y_2, \Delta y_3, \Delta y_4) \rightarrow (\Delta x_1^*, \Delta x_2^*). \quad (101)$$

Произвольный вектор ситуаций, построенный описанным способом, экстраполировали по двум ближайшим точкам обучающей выборки. Ошибку приближенного решения $(\Delta x_1, \Delta x_2)$ определяли по формуле

$$e = \sqrt{(\Delta x_1^* - \Delta x_1)^2 + (\Delta x_2^* - \Delta x_2)^2}.$$

По результатам 30 экспериментов средняя ошибка вычисления минимума была $\bar{e} = 0,124$. Эксперимент, в котором произвольную ситуацию экстраполировали по трем ближайшим ситуациям обучающей выборки, давал точное определение минимума функции по методу многомерной линейной экстраполяции.

Ниже, в примерах 8 и 9 исследуется более сложный случай, когда при поиске оптимального решения методом многомерной линейной экстраполяции используется информация не о самой функции, а о целом классе функций, заданном в обучающей выборке.

Пример 8. Рассмотрим квадратичную форму вида

$$y(x_1, x_2) = a_1(x_1 - x_1^*) + a_2(x_2 - x_2^*)^2 +$$

$$+ a_3(x_1 - x_1^*)(x_2 - x_2^*).$$

Чтобы в точке (x_1^*, x_2^*) достигался минимум функции, для коэффициентов a_1, a_2, a_3 должно выполняться условие

$$a_1 + a_2 > \sqrt{(a_1 a_2)^2 + a_3^2}.$$

Класс функции задается случайным выбором: коэффициентов a_1 и a_2 – из интервала $[0,5; 0,6]$, коэффициента a_3 – из интервала $[-0,05; 0,05]$ и значений x_1^* и x_2^* – из интервала $[0; 0,1]$. Значения x_1 и x_2 при каждой реализации функции взяты случайно из интервала $[0,35; 0,65]$. Обучающая выборка построена из 30 соответствий вида (101), генерированных случайно в указанных пределах. Каждая из 30 ситуаций в обучающей выборке экстраполирована по двум ближайшим ситуациям, в результате чего определена средняя ошибка вычисления $\bar{e} = 0,052$.

Пример 9. Класс кусочно-линейных функций вида

$$y = a |x - x^*| + b$$

задается случайным выбором: коэффициентов a и b – из интервала $[0,95; 1,05]$, x^* – из интервала $[0,45; 0,55]$. Начальные значения x взяты случайно из интервала $[0,35; 0,65]$. Обучающая выборка содержит 30 соответствий вида $(\Delta y_1, \Delta y_2) \rightarrow \Delta x^*$, где $\Delta x^* = x^* - x$ представляет собой оптимальное приращение аргумента, а $\Delta y_1 = y_1 - y_0$; $\Delta y_2 = y_2 - y_0$ есть приращения функции, полученные по трем измерениям конкретной функции из класса:

$$y_0 = f(x) = a |x - x^*| + b; y_1 = f(x + 0,2);$$

$$y_2 = f(x - 0,2).$$

Каждая из 30 ситуаций в обучающей выборке экстраполирована по k ближайшим ситуациям выборки. Определена средняя ошибка вычисления минимума. Для $k = 2$ получено $\bar{e} = 0,0017$; для $k = 3$ эта ошибка $\bar{e} = 0,0007$.

Рассмотренные примеры показывают, что метод многомерной линейной экстраполяции эффективен при решении задач поисковой оптимизации. Этот метод позволяет использовать информацию, полученную при решении поисковых задач из заданного класса. Учет локальных свойств обучающей выборки гарантирует большую точность определения решения по сравнению с регрессионными моделями.

4. ЭМПИРИЧЕСКОЕ ПРОГНОЗИРОВАНИЕ МЕТОДОМ МНОГОМЕРНОЙ ЛИНЕЙНОЙ ЭКСТРАПОЛЯЦИИ

Пусть протокол наблюдений (эмпирические данные) поведения объекта в экспериментах имеет вид

$$Pr = \langle X^i, Y^i, i = 1, \dots, N \rangle, \quad (102)$$

где $X = (x_1, \dots, x_m)$ – вход (причина), $Y = (y_1, \dots, y_p)$ – выход (следствие) наблюдаемого явления.

Между ними априори существует какая-то неизвестная детерминированная связь, т.е. векторная функциональная зависимость

$$Y = F(X). \quad (103)$$

Все показания (102) связаны этой зависимостью, т.е.

$$Y^i = F(X^i), i = 1, \dots, N.$$

Задача заключается в том, чтобы при отсутствии априорной информации о характере зависимости (103), используя эмпирические данные из протокола Pr (102), синтезировать эффективное правило (алгоритм) Φ^* прогноза значений выхода Y для любых значений входа X [34].

Определим понятие эмпирического правила Φ прогноза как последо-

вательность операций над эмпирическими данными (102), приводящую к некоторой оценке неизвестных значений выхода Y при заданных значениях входа X . Правило Φ характеризуется структурой S и параметрами C :

$$\Phi = \langle S, C \rangle.$$

Будем под структурой S понимать алгоритмические операции и порядок их выполнения при реализации правила, т.е. структуру его программы. Параметры $C = (c_1, \dots, c_5)$, необходимые для выполнения этой программы, являются параметрами правила. Очевидно, что качество q правила Φ зависит от его структуры и параметров

$$q = q(S, C).$$

Следовательно, задача создания эффективного прогнозирующего правила Φ^* заключается в решении задачи многокритериальной оптимизации

$$q(S, C) \rightarrow \underset{S \in \Xi, C \in \theta}{\text{extr}},$$

где Ξ — множество допустимых структур правила; θ — множество значений вектора параметров C , которое зависит от структуры S . При этом эффективное правило Φ^* должно удовлетворять протоколу (102), т.е. если \hat{y}^i есть оценка y^i для значений входа $X = X^i$ по правилу Φ^* , полученному на протоколе Pr [$\hat{y}^i = \Phi^*(X^i, \text{Pr})$], то

$$\hat{y}^i = y^i, i = 1, \dots, N.$$

Кроме того, правило Φ^* должно удовлетворять следующему важному требованию. Пусть имеется фрагмент протокола (102), т.е. подпротокол

$$\text{Pr}_1 = \langle X^i, Y^i, i \in \Omega_1 \rangle,$$

где Ω_1 — множество целых чисел отрезка из натурального ряда $\Omega = \{1, \dots, V\}$, т.е. $\Omega_1 \subset \Omega$. Дополнением этого подпротокола назовем подпротокол, содержащий множество $\bar{\Omega}_1$, для которого

$$\Omega_1 \wedge \bar{\Omega}_1 = 0;$$

$$\Omega_1 \vee \bar{\Omega}_1 = \Omega.$$

Это означает, что подпротокол и его дополнения не пересекаются и вместе образуют исходный протокол (102). Тогда естественно выдвинутое требование к эффективному правилу, чтобы Φ^* , полученное на базе подпротокола Pr_1 , хорошо прогнозировало дополнение этого подпротокола Pr_1 , т.е. если $\hat{y}^i = \Phi^*(X^i, \text{Pr}_1)$, то $\hat{y}^i \approx y^i$ для $i \in \bar{\Omega}_1$.

Задавая правило Φ_1 на множестве Ω_1 подпротокола Pr_1 наблюдений и меру точности его оценок Y^i ($i \in \bar{\Omega}_1$), которая может характеризоваться невязками

$$q_{1i} = |y^i - \Phi_1(X^i, \text{Pr}_1)|, i \in \Omega_1, \quad (104)$$

получаем характеристики эффективности решающего правила Φ_1 .

Теперь задача синтеза эффективного эмпирического правила прогноза сводится к отысканию такого правила Φ_1^* , которое минимизирует выражение (104). Многокритериальная задача минимизации (104) может

быть сведена к однокритериальной обычной сверткой критериев, например, в виде взвешенной суммы невязок

$$q_1 = \sum_{i \in \Omega_1} \alpha_{1i} q_{1i} \rightarrow \min_{\Phi_1} \Rightarrow \Phi_1^*, \quad (105)$$

где α_{1i} — весовые коэффициенты отдельных наблюдений при использовании правила Φ_1 .

Итак, для решения задачи (105) необходимо иметь следующие данные:

- 1) множество k подпротоколов Pr_1, \dots, Pr_k ;
- 2) правило свертки критериев (104);
- 3) множество допустимых структур Ξ ;
- 4) множество параметров θ каждой структуры.

Если структура правила S задана, то отпадает необходимость в задаче пункта 3.

Рассмотрим одно правило прогноза, заданное своей структурой. В основе этого правила лежит метод многомерной линейной экстраполяции [33]. Структура правила заключается в следующем. Правило задается выбором n пар $\{X^i, Y^i\}_{i=1}^n$ из всех Ω_1 пар подпротокола (класса) Pr_1 . Для каждого нового значения входа X^* определяется принадлежность его к одному из k подпротоколов в пространстве X с метрикой $\rho(X^i, X^r)$, определяющей расстояние между двумя любыми входами X^i и X^r . Вектор X^* отождествляется с вектором \hat{X}^* , являющимся линейной комбинацией $n < N$ отобранных векторов X^i :

$$\hat{X}^* = \sum_{i=1}^n \lambda_i^* X^i,$$

где $\Lambda^* = (\lambda_1^*, \dots, \lambda_n^*)$, $\sum_{i=1}^n \lambda_i^* = 1$.

Это осуществляется решением задачи наилучшего линейного приближения X^* вектором \hat{X}^* в смысле минимума $\|X^* - \hat{X}\|$, т.е.

$$\|X^* - \sum_{i=1}^n \lambda_i X^i\| \rightarrow \min_{\Lambda} \Rightarrow \Lambda^*.$$

Полученное значение Λ^* используется для оценки Y^* :

$$\hat{Y}^* = \sum_{i=1}^n \lambda_i^* Y^i.$$

Очевидно, что точность оценок \hat{Y}^* , полученная настоящим правилом, будет зависеть от метрики ρ и от соответственно ей выбранных подпротоколов Pr_1 (классов), а также выбранных n пар из Ω_i пар $(i = 1, \dots, k)$ для синтеза правила. Это и есть параметры S правила прогноза. Простейшим заданием метрики является евклидово расстояние с весами β_j ($j = 1, \dots, m$) составляющих вектора X . Весовые коэффициенты могут быть заданы либо экспертно, либо получены в процессе обучения минимизацией ошибки прогноза по заданному правилу Φ^* и Pr . При заданных метрике ρ и подпротоколе $Pr_1 (i = 1, \dots, k)$ выбор эффективных правил $\Phi_1^* (i = 1, \dots, k)$ для каждого подпротокола осуществляется решением задачи (105). Весовые коэффициенты α_{1i} выбирают исходя из удаленности i -го вектора X^i от \hat{X}^i : чем больше $\|X^i - \hat{X}^i\|$ в заданной метрике ρ , тем меньше должен быть вес α_{1i} .

Предлагаемый алгоритм прогноза применен при предсказании значений показаний одного из приборов (вольтметра, $p = 1$) по показаниям двух других (амперметра и омметра, $m = 2$) для участка электрической цепи [6]. В качестве эмпирических данных использовались показания всех трех приборов для других участков цепи. Эксперименты осуществлялись двояким образом.

1. Задавалась евклидова метрика с весами $\beta_j = 1/m$ ($j = 1, \dots, m$). Для каждого i -го наблюдения ($i = 1, \dots, N; N = 16$) задавался свой локальный подпротокол P_{i1} числом T ($T = 3, \dots, 7$) ближайших к X^i входов X^r ($r = 1, \dots, N; r \neq i$) и соответствующих выходов Y^r , т.е. парами $\{X^r, Y^r\}_{r=1}^T$. Правила генерировались двумя способами:

1) перебором всех комбинаций из T по n (n — эксперименте задавалось равным 2); в этом случае в качестве правила Φ_1^* выбирали то, которое имело наименьшее значение q_1 при решении задачи (105);

2) в качестве Φ_1^* выбирали комбинацию $\{X^{r*}, Y^{r*}\}_{r=1}^T$ с $|Y^r - Y^t| \rightarrow \max_{r, t} (r, t = 1, \dots, T)$.

Полученные в обоих случаях правила Φ_1^* применяли для прогноза Y^i .

Усредняли значение модуля ошибки $\delta = |Y^i - \hat{Y}^i|$ по всем $N = 16$ наблюдениям. Результаты для различных значений T при выборе правила Φ_1^* по способу 1

T	...	3	4	5	6	7
Φ_1^*	...	529	476	407	521	610

по способу 2

T	...	3	4	5	6	7
Φ^*	...	495	427	501	542	587

В обоих случаях зависимости $\bar{\delta} = f(T)$ носят экстремальный характер, и экстремальные значения T^* могут использоваться для выбора локального подпротокола при прогнозе Y^* для новых значений входа X^* .

2. Во втором эксперименте определяли оптимальную евклидову метрику (весовые коэффициенты β_1 и β_2 параметров X_1 и X_2). В качестве правила прогноза Φ_1^* выбирали правило, построенное по двум ближайшим к X^i парам $\{X^r, Y^r\}_{r=1}^2$. Оптимальными параметрами метрики выбирали такие значения β_1 и β_2 , которые на данном протоколе ($N = 16$) и при выбранном правиле Φ_1^* обеспечивают минимум эмпирического риска (значение средней ошибки прогноза). На β_1 и β_2 накладывали ограничения нормировки $\beta_1 + \beta_2 = 1; \beta_1, \beta_2 \geq 0$. Были получены следующие значения: $\beta_1^* = 0,38; \beta_2^* = 0,62$ (при этом значение средней ошибки прогноза равно 499, в то время как при $\beta_1 = \beta_2 = 0,5$ эта величина равна 589).

5. МЕТОД МНОГОМЕРНОЙ ЛИНЕЙНОЙ ЭКСТРАПОЛЯЦИИ В ЗАДАЧАХ ВЫЯВЛЕНИЯ ДИНАМИЧЕСКИХ ЗАКОНОМЕРНОСТЕЙ

Поведение всякой динамической системы характеризуется соотношением между векторами входа X_i и состояния системы Y_i в дискретном времени i :

$$\langle X_i, Y_i \rangle \quad (i = 1, \dots, N), \quad (106)$$

где $X_1 = (x_{1i}, \dots, x_{ni})$; $Y_1 = (y_{1i}, \dots, y_{mi})$;

n — число наблюдаемых входов объекта; m — число наблюдаемых выходов объекта.

Задачу обнаружения динамической закономерности естественно представить как задачу синтеза модели объекта в виде [24]

$$Y = F(X),$$

где F — искомый оператор.

В простейшем виде динамический объект описывается дифференциальным уравнением

$$dY/dt = f(X, Y),$$

где f — неизвестная функция.

Для дискретного случая имеем

$$Y_i = Y_{i-1} + F(X_i, Y_{i-1}).$$

Запишем изменение состояния объекта в виде $Z_i = F(X_i, Y_{i-1})$, т.е. $Z_i = Y_i - Y_{i-1}$.

Из соотношения (106) можно получить последовательность изменений состояния объекта

$$\langle Z_i, i = \overline{1, N} \rangle, \quad Z_i = (z_{1i}, \dots, z_{mi}).$$

Теперь, располагая информацией

$$I = \langle X_i, Y, Z_i, i = \overline{1, N} \rangle,$$

можно определить оценки выхода объекта Y_k' для другой заданной входной последовательности $X_k', k = \overline{1, M}$, отличной от $X_i, i = \overline{1, N}$:

$$\hat{Y}'_l = \hat{Y}_{l-1} + \varphi(1, X'_{l-1}, \hat{Y}'_{l-1}), \quad (l = \overline{1, M}),$$

где X_0', Y_0' — начальные условия, а φ — алгоритм оценки изменения состояния системы:

$$\hat{Z}'_l = \varphi(1, X'_{l-1}, \hat{Y}'_{l-1}).$$

В качестве алгоритма φ предлагается использовать метод многомерной линейной экстраполяции, позволяющей оценить Z'_l [35].

В данном параграфе использована следующая модификация этого метода. Опорная последовательность образуется информацией I в виде

$$\langle U_i, Z_i \rangle \quad (i = \overline{1, N}), \quad (107)$$

где U_i — $n + m$ -мерный вектор вида

$$U_i = (X_i, Y_i).$$

Искомую оценку Z'_l определим как линейную экстраполяцию опор-

ной последовательности по U_{l-1} . Для этого выбираем из опорной последовательности (107) k векторов, ближайших к U_{l-1} , и образуем их линейную комбинацию

$$U = \sum_{i=1}^k \lambda_i U_i.$$

Коэффициенты λ_i определяем так, чтобы расстояние между U и U_{l-1} было минимальным, т.е. решаем задачу

$$\rho = \left| \sum_{i=1}^k \lambda_i U_i - U_{l-1} \right| \rightarrow \min_{\lambda_i}.$$

Полученные при этом значения $\lambda_1^*, \dots, \lambda_k^*$ и дают возможность оценить Z_l' по формуле

$$\hat{Z}_l' = \sum_{i=1}^k \lambda_i^* Z_i,$$

где Z_i ($i = \overline{1, k}$) — соответствующие значения Z из опорной последовательности (107).

Для иллюстрации работоспособности изложенной методики решалась следующая модельная задача. Динамический объект моделировался дифференциальным уравнением

$$dy/dt = a_1 y + a_2 x_1 + a_3 x_2 + a_4,$$

т.е. рассматривался объект, имеющий два входа $X = (x_1, x_2)$, ($n = 2$) и выход $Y = y$, ($m = 1$). В качестве входных воздействий были приняты следующие функции:

$$x_1 = f_1(t) = \sin \omega_1 t;$$

$$x_2 = f_2(t) = \sin \omega_2 t; \omega_2 = c \omega_1.$$

Моделирование обучающей выборки (106) проводилось при следующих значениях коэффициентов: $a_1 = -1$; $a_2 = a_3 = a_4 = 1$; $\omega_1 = 2$; $c = 1,1$. Поведение входов x_1, x_2 и выхода y представлено на рис. 28 и 29 (значение дискретности $t = 0,1$; $N = 31$).

Новое поведение входа X задавалось при значениях $\omega_1 = 1$ и $c = 1,2$

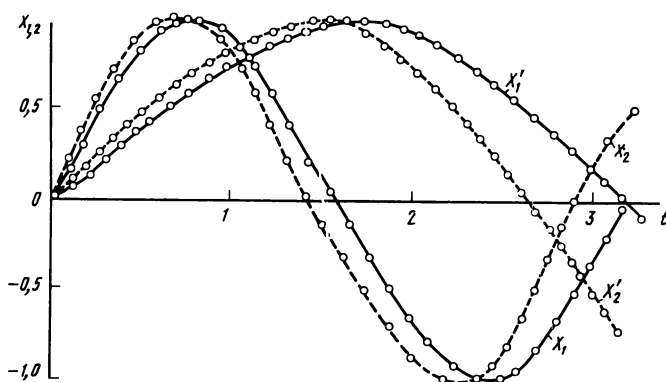


Рис. 28. Обучающая выборка (X_1 и X_2) и контрольное возмущение (X_1', X_2')

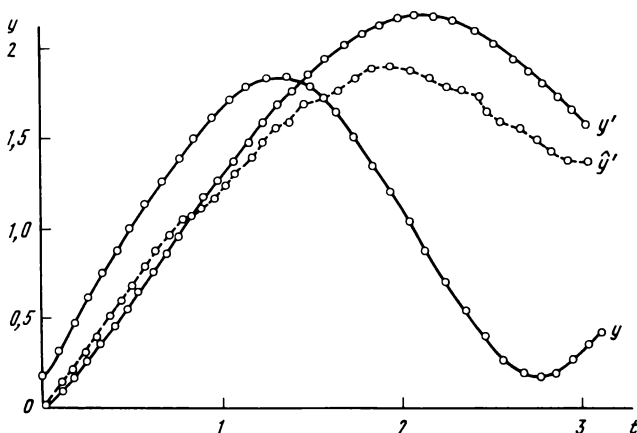


Рис. 29. Поведение динамического объекта на обучающей выборке (y), оцененное (\hat{y}') и точное (y')

(на рис. 28 $M = 32$). После применения изложенной методики для определения оценок \hat{y}' полученный результат оказался близок к точному y' (рис. 29). Значительное отклонение от y' во второй половине процесса связано с тем, что обучающая выборка здесь значительно отличается от того, что требуется, и "близость" здесь весьма отдаленная, что и привело к возрастанию ошибки.

6. ПОИСКОВЫЕ АЛГОРИТМЫ ОПРЕДЕЛЕНИЯ МНОЖЕСТВА ПАРЕТО

Задача адаптации сложных систем неизбежно связана с необходимостью рассмотрения большого числа критериев, оптимизация которых и составляет задачу адаптации. Первым шагом для решения такой многокритериальной задачи адаптации является надежный выход на компромиссное множество (множество Парето), внутри которого расположено решение. Эта задача представляет определенные алгоритмические трудности, преодоление которых является задачей данного исследования [36].

Постановка задачи. Рассмотрим задачу многокритериальной оптимизации

$$q_i(X) \rightarrow \min_{X \in S} (i = 1, \dots, k), \quad (108)$$

где $q_i(\cdot)$ — скалярная функция векторного аргумента, $X = (x_1, x_2, \dots, x_n)$, а S — область его определения; $h_l(X) \geq 0$ ($l = 1, m$). Как известно [37], она решается из множества Парето (P — множество) неулучшаемых точек X^* , удовлетворяющих условию

$$q_i(X^*) \leq q_i(X) (i = 1, \dots, k), \quad (109)$$

где $X^* \in P \subset S$; $X \in S$. Условие (109) выражает следующее очевидное свойство этого множества: для каждого $X \in S$ всегда имеется $X^* \in S$,

удовлетворяющее неравенству (109), причем для X^* не найдется улучшаемой по (109) точки.

Для определения точек X^* могут быть применены поисковые методы оптимизации, модифицированные для применения к векторной функции качества. В этом случае алгоритмы выхода на множество Парето представляют собой итеративные процедуры, на первом этапе которых определяется вектор направления спуска V ($\|V\| = 1$), удовлетворяющий системе неравенств

$$q_i(X) \geq q_i(X + aV) \quad (i = 1, \dots, k), \quad (110)$$

где a — достаточно малая положительная величина.

На втором этапе производится спуск вдоль выбранного направления V до минимума одной из функций (108) при $X \in S$ или до выхода на границу области S , т.е. при $X \in S$, где S — граница S .

При решении последней задачи могут быть использованы обычные методы одномерной оптимизации, например, метод золотого сечения [37].

В данной работе рассмотрены два алгоритма, реализующие выход на множество Парето и поиск в этом множестве. Главное внимание при их разработке уделено первому этапу определения допустимого направления спуска.

Описание алгоритмов. *Алгоритм 1.* При определении направления спуска V , удовлетворяющего условию (110), можно воспользоваться локальными характеристиками критериев — их градиентами $\text{grad } q_i(X_j)$ ($i = 1, \dots, k$) в исходной точке X_j . В этом случае условие (110) эквивалентно требованию, чтобы все проекции $R_i(V)$ ($i = 1, \dots, k$) вектора V на антиградиентные направления критериев $\text{dir}(-\text{grad } q_i(X_j))$ ($i = 1, \dots, k$) были положительны или равны нулю, т.е. необходимо выполнение следующей системы неравенств:

$$R_i(V) \geq 0 \quad (i = 1, \dots, k), \quad (111)$$

где $R_i(V) = V, \text{dir}(-\text{grad } q_i(X_j))$. Назовем вектор V , удовлетворяющий условию (111), допустимым.

В целях улучшения последующего спуска вдоль выбираемого направления V целесообразно это направление выбрать таким образом, чтобы оно находилось в центре пучка допустимых векторов. Это означает, что поиск наилучшего допустимого вектора V^* следует производить так, чтобы минимальная проекция (111) была максимальной. Следовательно, для определения оптимального направления спуска V^* необходимо решение задачи

$$\begin{aligned} \min R_i(V) &\rightarrow \max_{V \in D} \\ D: \int R_i(V) &\geq 0 \quad (i = 1, \dots, k), \\ &\|V\| = 1. \end{aligned} \quad (112)$$

Возможно несколько подходов при решении задачи (112). Наиболее предпочтительным представляется использование методов линейного программирования. Для этого необходимо:

1) перейти к новой системе координат, в которой компоненты вектора направления принимали бы только положительные значения;

- 2) представить функцию качества задачи (112) в линейном виде;
- 3) преобразовать ограничение $\|V\| = 1$ в линейную форму.

После выполнения указанных требований задача (112) принимает следующий вид:

$$\delta \rightarrow \max_{V'; \delta \in D_1} \quad (113)$$

$$D_1: \begin{cases} [(V' - I), \text{dir}(-\text{grad } q_i(X_j))] - \delta \geq 0 & (i = 1, \dots, k) \\ 0 \leq V_r' \leq 2 & (r = 1, \dots, n), \end{cases}$$

где $V' = (v_1', v_2', \dots, v_n')$, $V' = V + 1$, $I = (1, 1, \dots, 1)$.

Решение задачи (113) можно получить с помощью симплекс-метода.

Влияние ограничений $h_l(X) \geq 0$ ($l = 1, \dots, m$), определяющих допустимую область S изменения параметров X , при поиске оптимального направления спуска V^* можно учесть вводя дополнительные неравенства $R_{k+z}(V) = V, \text{dir}(\text{grad } h_z(X_j)) > 0$ ($z = 1, \dots, m'$) в систему (111), где m' — число существенных ограничений, которые в данной точке X_j необходимо учитывать.

При больших размерностях задачи и в случаях, когда критерии q_i ($i = 1, \dots, k$) представляют собой сложные овражные функции, более целесообразно при выборе оптимального направления спуска V^* воспользоваться аппаратом случайного поиска.

Алгоритм 2. Рассмотрим в качестве такого подхода алгоритм случайного поиска на сфере. Он сводится к следующему. В исходной точке X_j определяется сфера достаточно малого радиуса a . На ней случайным образом выбирается точка X_g и генерируется случайное направление по сфере.

Если в этом направлении на сфере имеются точки более предпочтительные, чем X_g , то производится спуск в наилучшую точку X_{g+1} , в противном случае выбирается новое случайное направление.

При невозможности дальнейшего улучшения последней точки X_{g+p} , где p — число спусков по сфере, и выполнении системы неравенств

$$q_i(X_j) \geq q_i(X_{g+p}) \quad (i = 1, \dots, k), \quad (114)$$

когда по крайней мере одно из неравенств выполняется строго, оптимальное направление спуска

$$V^* = \text{dir}(X_{g+p} - X_j).$$

Если условие (114) не удовлетворяется, то допустимого вектора V не существует, и точка X_j принадлежит множеству Парето.

Анализ различных ситуаций определения V^* на сфере показывает, что возможны ситуации, в которых отсутствуют случайные точки в районе исходной X_g . В связи с этим поиск оптимального направления необходимо разделить на три этапа.

1. Выход в зону на сфере, где возможно формирование некоторой унимодальной функции Q_2 (см. ниже), поиск минимума которой экви-

валентен нахождению допустимого направления V . Это можно осуществить, например, в результате решения задачи

$$Q_1 = q_r(X) \rightarrow \min_{X \in T},$$

где T — множество точек на поверхности сферы, причем выбор номера r осуществляется из условия

$$q_r(X_g) - q_r(X_j) = \max (q_i(X_g) - q_i(X_j)),$$

т.е. для критерия, имеющего максимальную разность на этой сфере. Опыт показывает, что в 90 % случаев эта минимизация дает направление допустимого вектора.

2. Поиск допустимого направления. В качестве Q_2 можно выбрать функцию следующего вида:

$$Q_2 = \sum_{i=1}^k [(q_i(X) - q_i(X_j)) \varphi(q_i(X), q_i(X_j))],$$

где $X \in T$ и

$$\varphi(q_i(X), q_i(X_j)) = \begin{cases} 1 & \text{при } q_i(X) \geq q_i(X_j), \\ 0 & \text{при } q_i(X) < q_i(X_j). \end{cases}$$

Если $Q_2 = 0$, то $V = \text{dir}(X - X_j)$ является допустимым направлением.

3. Определение оптимального направления спуска V^* :

$$V^* = \text{dir}(X^* - X_j),$$

где

$$X^* = \arg \max_{X \in T_1} \min_{(i=1, \dots, k)} \left| \frac{q_i(X) - q_i(X_j)}{q_i(X_j)} \right|,$$

T_1 — множество точек на сфере, определяющих допустимые направления.

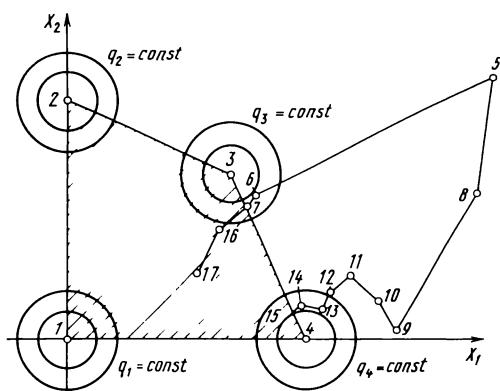
Влияние ограничений $h_l(X) \geq 0$ ($l = 1, \dots, m$) в данном случае можно учесть введением дополнительных функций штрафа в выражения q_i ($i = 1, \dots, k$) или в критерии предпочтительности точек на поверхности сферы на всех трех этапах поиска оптимального направления спуска.

При адаптивном подходе [18] решения задач векторной оптимизации требования лица, принимающего решения (ЛПР), поступают в различных формулировках. Наиболее часто указывают следующие множества критериев:

- 1) значения которых необходимо улучшить ($\mathcal{P}_1 \subset \{q_1, q_2, \dots, q_k\}$)
- 2) значения которых не должны ухудшаться по отношению к рассматриваемой ситуации ($\mathcal{P}_2 \subset \{q_1, q_2, \dots, q_k\}$);
- 3) значения которых на данном этапе несущественны ($\mathcal{P}_3 \subset \{q_1, q_2, \dots, q_k\}$):

Для учета пожеланий ЛПР на отдельных итерациях процесса решения многокритериальной задачи, т.е. для перехода на более предпочтительную (по мнению ЛПР) точку множества Парето, могут быть использованы рассмотренные выше алгоритмы. В этом случае критерии, составляющие

Рис. 30. Поиск точек на множестве Парето P (заштрихованная область) с использованием алгоритма 1 и 2 при $P_1 = \{q_1, q_2, q_3, q_4\}$, $P_2 = P_3 = \phi$ — траектории 5–7, при $P_1 = \{q_1\}$, $P_2 = \{q_4, q_2\}$, $P_3 = \{q_3\}$ — траектории 7, 16, 17 и с применением алгоритма 3 при $P_1 = \{q_1, q_2, q_3, q_4\}$, $P_2 = P_3 = \phi$ — траектории 5–15



множество \mathcal{P}_2 , образуют дополнительное семейство ограничений, а влияние критериев из множества \mathcal{P}_3 не рассматривается.

Эксперименты и результаты. Рассмотренные алгоритмы для определения точек множества Парето были апробированы на нескольких модельных задачах при размерностях пространства параметров $2 \leq n \leq 6$ и с числом критериев $k = 4$. Для оценки работы алгоритмов с использованием антиградиентных направлений (алгоритм 1) и случайного поиска по сфере (алгоритм 2) использовался базисный алгоритм (алгоритм 3), в котором допустимое направление спуска определялось перебором случайных направлений.

На рис. 30 показаны траектории поиска точек множества Парето указанными алгоритмами при $n = 2, k = 4$.

Сопоставление потерь на поиск алгоритмов 1–3 приведено в табл. 14, где представлены отношения этих потерь, показывающие, во сколько раз алгоритмы 1 и 2 лучше, чем алгоритм 3. Хорошо видна высокая эффективность этих алгоритмов.

Т а б л и ц а 14

Размерность пространства параметров	2	3	5	6
Число решенных задач	7	5	4	4
Отношения потерь на поиск алгоритмов 3 и 1	7,2	8,1	8,7	8,9
Отношения потерь на поиск алгоритмов 3 и 2	2,1	3,6	5,3	5,7

7. ФИЛЬТР КАЛМАНА И МЕТОД МНОГОМЕРНОЙ ЭКСТРАПОЛЯЦИИ

Известно, что при построении фильтра Калмана необходимо задаться моделью наблюдаемого сигнала. Многочисленные публикации, посвященные калмановской фильтрации, этот вопрос, как правило, оставляют вне сферы своего внимания. Насколько удачно выбрана модель наблюдаемого сигнала — зависит от научной эрудиции исследователя, от степени

проникновения его в сущность изучаемого явления и т.д. И порой на практике эта часть задачи оказывается самой сложной, поскольку все остальные вопросы, связанные с построением фильтра Калмана, исследованы достаточно подробно. Чем удачнее выбрана модель, тем, в общем случае, выше качество фильтрации фазовых координат наблюдаемого процесса. Особенно заметное влияние выбор модели оказывает на качество фильтрации в переходных режимах, когда реальный процесс резко изменяет свои параметры. И вполне естественно, что направление исследования фильтра Калмана связано с разработкой вопроса выбора моделей наблюдаемого сигнала.

В данном параграфе в постановочной форме показана возможность решения задачи выбора модели сигнала методом многомерной экстраполяции. Покажем это формально [23].

Пусть линейный стохастический многошаговый процесс описывается уравнениями [4]

$$x_{i+1} = \Phi_i x_i + \Gamma_i w_i, \quad i = 0, \dots, N-1; \quad (115)$$

$$E[x_0] = \bar{x}_0;$$

$$E[w] = \bar{w}_i;$$

$$E[(x_0 - \bar{x}_0)(x_0 - \bar{x}_0)^T] = M_0;$$

$$E[(w_i - \bar{w}_i)(w_j - \bar{w}_j)^T] = Q\delta_{ij};$$

$$E[(w_i - \bar{w}_i)(x_0 - \bar{x}_0)^T] = 0.$$

Когда система находится в состоянии i , проводятся измерения z_i , линейно связанные с состоянием x_i :

$$z_i = H_i x_i + v_i, \quad i = 0, \dots, N;$$

$$E[v_i] = 0;$$

$$E[v_i v_j^T] = R_i \delta_{ij};$$

$$E[(w_i - \bar{w}_i) v_j^T] = 0 \text{ и } E[(x_0 - \bar{x}_0) v_i^T] = 0.$$

В этом случае оценку состояния x_k можно записать в виде

$$\hat{x}_i = \bar{x}_i + k_i (z_i - H_i \bar{x}_i), \quad i = 0, \dots, k, \quad k \leq N,$$

где

$$\bar{x}_{i+1} = \Phi_i \hat{x}_i + \Gamma_i \bar{w}_i, \quad x_0 \text{ задано,}$$

$$k_i = P_i H_i^T R_i^{-1},$$

$$P_i = (M_i^{-1} + H_i^T R_i^{-1} H_i)^{-1} = M_i - M_i H_i^T (H_i M_i H_i^T + R_i)^{-1} H_i M_i,$$

$$M_{i+1} = \Phi_i P_i \Phi_i^T + \Gamma_i Q_i \Gamma_i^T.$$

Это и есть фильтр Калмана для линейных многошаговых процессов. Для его построения необходимо задаться уравнением (115). Покажем работу фильтра для следующего примера.

Пусть модель сигнала выбрана в виде двух "сшитых" парабол (рис. 31, кривая 1):

$$\begin{cases} x = At^2 + Bt + C, & t \leq t_1, \\ x = -At^2 - Bt - C, & t > t_1. \end{cases}$$

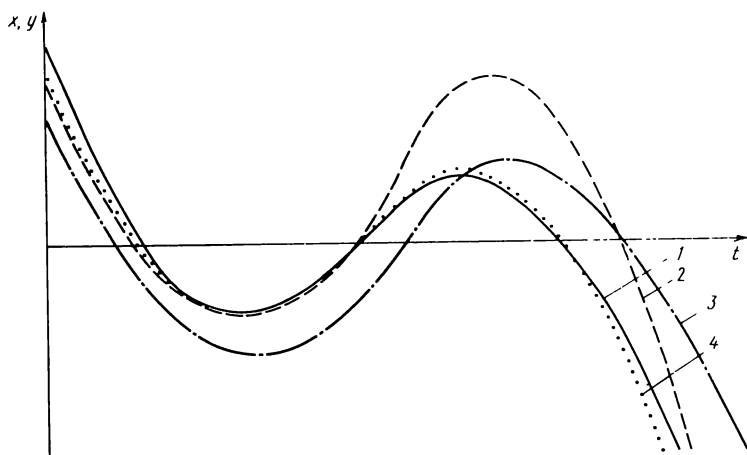


Рис. 31. Модель полезного сигнала (кривая 1); оценка сигнала на выходе фильтра Калмана (кривая 2); оценка, полученная методом многомерной экстраполяции (кривая 3); оценка сигнала, полученная на выходе синтезированного фильтра (кривая 4)

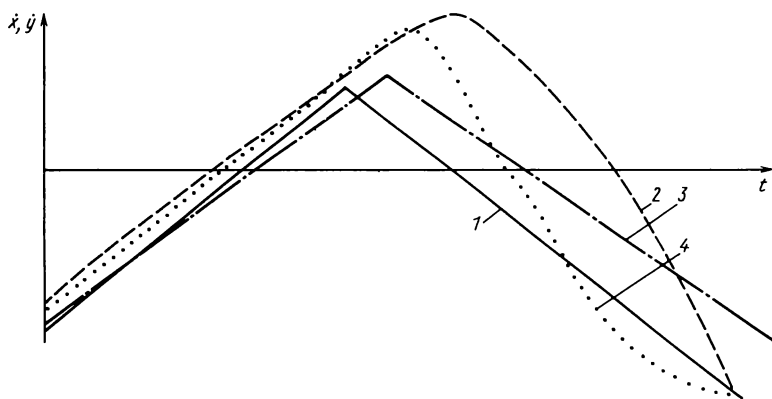


Рис. 32. Первая производная полезного сигнала (кривая 1), оценка производной на выходе фильтра Калмана (кривая 2); оценка производной, полученная методом многомерной экстраполяции (кривая 3); оценка производной, полученная на выходе синтезированного фильтра (кривая 4)

На рис. 32 и 33 кривые 1 — первая и вторая производные этого сигнала соответственно. Из анализа рис. 33 видно, что вторая производная сигнала терпит разрыв в точке t_1 . В данном случае матрицу Φ естественно выбрать следующим образом:

$$\begin{aligned} \dot{x}^1 &= x^2; \\ \dot{x}^2 &= x^3, \Rightarrow \Phi = \begin{vmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{vmatrix}, H = \begin{vmatrix} 1 \\ 0 \\ 0 \end{vmatrix} \\ \dot{x}^3 &= 0. \end{aligned}$$

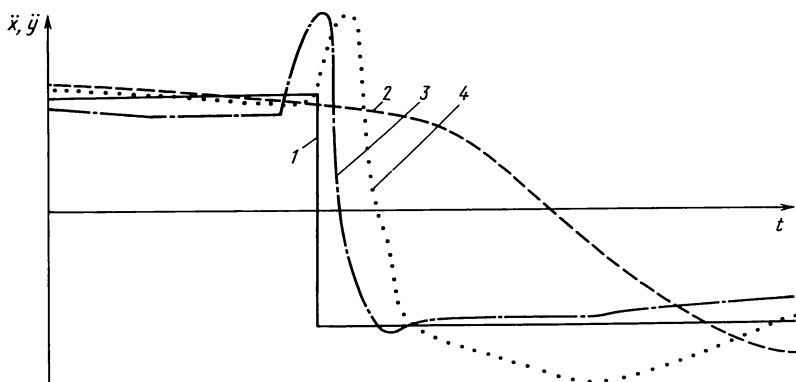


Рис. 33. Вторая производная полезного сигнала (кривая 1); оценка ее на выходе фильтра Калмана (кривая 2); оценка, полученная методом многомерной экстраполяции (кривая 3); оценка, полученная на выходе синтезированного фильтра (кривая 4)

Тогда фильтр Калмана можно записать в виде

$$\hat{x}_i^1 = \bar{x}_i^1 + k_1 (z - \bar{x}_i^1);$$

$$\hat{x}_i^2 = \bar{x}_i^2 + k_2 (z - \bar{x}_i^1);$$

$$\hat{x}_i^3 = \bar{x}_i^3 + k_3 (z - \bar{x}_i^1).$$

С учетом перехода от непрерывного наблюдения к дискретному значения \bar{x}_i будут

$$\bar{x}_{i+1}^1 = \hat{x}_i^1 + \hat{x}_i^2 \Delta t + \Gamma_i \Delta \bar{w}_i;$$

$$\bar{x}_{i+1}^2 = \hat{x}_i^2 + \hat{x}_i^3 \Delta t;$$

$$\bar{x}_{i+1}^3 = \hat{x}_i^3.$$

(116)

На рис. 31 – 33 зависимости 2 показывают качество оценки фильтром Калмана выбранного полезного сигнала и его производных. Из анализа этих зависимостей видно, что фильтр Калмана работает с высокой точностью до момента t_1 , т.е. до момента смены знака второй производной. В точке t_1 вторая производная сигнала скачкообразно меняет знак. На такое изменение второй производной фильтр Калмана не может отреагировать достаточно быстро, что приводит к значительным ошибкам в оценке сигнала его производных.

Попытаемся воспользоваться для решения поставленной задачи методом многомерной экстраполяции. Этот метод обеспечивает высокую точность, если удачно выбраны обучения. В нашем случае в качестве обучений выбирается "пучок" парабол, на основании которого строим матрицу априорной информации

$$S = \begin{bmatrix} s_{11} & s_{12} & \dots & s_{1n} \\ s_{21} & s_{22} & \dots & s_{2n} \\ \dots & \dots & \dots & \dots \\ s_{m1} & s_{m2} & \dots & s_{mn} \end{bmatrix}.$$

В этой матрице каждая строка соответствует данным об одном про-

цессе, близком к заданному. Далее с помощью псевдообратной матрицы s^+ находим вектор

$$f_i = s^+ z_i,$$

минимизирующий

$$\Delta = \|z_i - s f_i\|.$$

На основании этого рекуррентная форма метода многомерной экстраполяции будет иметь следующий вид:

$$y_{i+1} = s s^+ y_i.$$

Качество работы фильтра, построенного методом многомерной экстраполяции, показано на рис. 31 – 33 в виде зависимостей 3. Анализ этих зависимостей показывает, что метод многомерной экстраполяции значительно превосходит фильтр Калмана по точности оценки координат сигнала в переходном режиме. Что касается установившегося режима, то здесь фильтр Калмана имеет бесспорные преимущества. Отсюда следует вполне реальная возможность улучшения качества работы фильтра Калмана в переходном режиме с помощью метода многомерной экстраполяции.

Из анализа зависимости 3, представленной на рис. 33, видно, что разность $\mu = y_{i+1}^3 - y_i^3$ мала до и после переключения, а в момент переключения она равна значению скачка второй производной сигнала. Перепишем третье уравнение системы (116) в виде

$$x x_{i+1}^3 = x_i^3 + \mu.$$

Такое добавление μ до и после скачка не изменит фильтра Калмана из-за малости этой величины. А в момент скачка μ резко возрастает, что равносильно переходу фильтра на новые начальные условия по второй производной. Качество работы такого фильтра в виде зависимостей 4 представлено на рис. 31 – 33. Анализ этих зависимостей показывает, что синтезированный фильтр имеет достаточно высокую точность оценки координат сигнала как в установившемся, так и в переходном режиме.

Таким образом, идеи многомерной экстраполяции могут с успехом использоваться для улучшения работы фильтра Калмана в переходном режиме.

Г Л А В А IV ПУТИ СОВЕРШЕНСТВОВАНИЯ АЛГОРИТМА МНОГОМЕРНОЙ ЭКСТРАПОЛЯЦИИ

1. РЕКУРРЕНТНАЯ ФОРМА МЛЭ

Существующие методы оптимизации сложных систем, как правило, связаны с решением громоздкой и поэтому трудоемкой задачи математического программирования. Метод многомерной линейной экстрапо-

ляции отличается именно тем, что позволяет приближенно решать задачу оптимизации без существенных затрат. Это обстоятельство и делает его очень перспективным при решении оптимизационных задач.

Главное преимущество МЛЭ сводится к тому, что он позволяет достаточно надежно решать задачу восстановления функций в условиях информационной недостаточности (малое число наблюдаемых точек и малое время наблюдения, характерное для реальной действительности). Рекуррентная форма метода, рассматриваемая здесь, позволяет вести вычисления в пространствах большой размерности без увеличения необходимой оперативной памяти ЭВМ, что важно при решении широкого круга задач проектирования, управления и т.д.

В основу МЛЭ положена минимизация разности

$$\rho^2 = \|Z - HX\|^2 \Rightarrow \min, \quad (117)$$

где Z — вектор, характеризующий наблюдаемую ситуацию; H — матрица, составленная из векторов, характеризующих обучающие ситуации; X — вектор весовых коэффициентов.

Другими словами, разность (117) характеризует отклонение линейной комбинации HX от ситуации, описанной вектором Z . Поставленная задача сводится к решению уравнения $Z = HX$, что в свою очередь требует существования матрицы H .

Как показывает опыт решения многочисленных задач, далеко не во всех случаях матрица H^{-1} существует. Речь идет о случаях, когда некоторые векторы, образующие матрицу H , оказываются близки к линейно зависимым. Для решения подобного рода задач целесообразно использовать псевдообратную матрицу H^+ , которая в отличие от обратной существует всегда. Рассмотрим несколько примеров, поясняющих существование псевдообратной матрицы.

Пусть требуется минимизировать $\|Z - HX\|^2$, когда размерности векторов Z и X равны 1. В этом случае матрица H имеет размер 1×1 и является просто скаляром. Поставленная задача сводится к нахождению числа x , минимизирующего величину $(z - hx)^2$, а если таких чисел будет много, то требуется из них выбрать число, имеющее минимальную норму (заметим, что для скаляров минимальная норма совпадает с их квадратом).

Решение элементарно. Если $h \neq 0$, то $x = z/h$ — единственное значение, минимизирующее эту функцию. Если же $h = 0$, то эту функцию минимизирует любое значение x , т.е. в этом случае функция от x не зависит. А среди всех чисел минимальным квадратом обладает, конечно, ноль, т.е. если $h = 0$, то $x = 0$.

В итоге, обозначая

$$h^+ = \begin{cases} 1/h, & \text{если } h \neq 0; \\ 0, & \text{если } h = 0, \end{cases}$$

получим решение в виде

$$x = h^+ z.$$

На этом простейшем примере мы показали динамику формирования псевдообратной матрицы размера 1×1 .

Пусть теперь $\mathbf{H} \in L_{2,2}$, причем матрица имеет диагональный вид, т.е.

$$\mathbf{H} = \begin{pmatrix} h_1 & 0 \\ 0 & h_2 \end{pmatrix}.$$

Тогда

$$\mathbf{H}^+ = \begin{pmatrix} h_1^+ & 0 \\ 0 & h_2^+ \end{pmatrix},$$

где h_1^+ и h_2^+ определяются, как и в предыдущем примере.

Рассмотрим еще один случай, когда матрица \mathbf{H} представляет собой вектор-столбец

$$\mathbf{H} = (h_1, h_2, \dots, h_n)^T,$$

и получим для этого случая псевдообратную матрицу. Элементы этой матрицы могут быть вычислены по формуле

$$a_i = \begin{cases} h_i / \mathbf{H}^T \mathbf{H}, & \text{если } \mathbf{H}^T \mathbf{H} \neq 0; \\ 0, & \text{если } \mathbf{H}^T \mathbf{H} = 0. \end{cases} \quad (118)$$

Здесь $\mathbf{H}^T \mathbf{H}$ — норма вектора \mathbf{H} ; если она не равна нулю, то на нее можно делить.

Рассмотрим формирование псевдообратной матрицы на следующем примере. Пусть $\mathbf{H} = (1; 2)^T$. Тогда

$$\mathbf{H}^T \mathbf{H} = (1; 2) \cdot (1; 2)^T = 1 \cdot 1 + 2 \cdot 2 = 5.$$

Используя формулу (118), формируем псевдообратную матрицу

$$a = 1/5 = 0,2, \quad a_2 = 2/5 = 0,4;$$

$$\mathbf{H}^+ = (0,2; 0,4).$$

Убедиться в том, что полученная матрица является действительно псевдообратной, можно с помощью условий Пенроуза [3]:

1. $\mathbf{H}^+ \mathbf{H}$ и $\mathbf{H} \mathbf{H}^+$ — симметричные матрицы;
2. $\mathbf{H} \mathbf{H}^+ \mathbf{H} = \mathbf{H}$, $\mathbf{H}^+ \mathbf{H} \mathbf{H}^+ = \mathbf{H}^+$.

Проверка этих условий показывает, что полученная в последнем примере матрица является действительно псевдообратной.

Теперь рассмотрим рекуррентный способ вычисления псевдообратных матриц произвольной размерности.

Пусть $a_1, a_2, a_3, \dots, a_m$ — вектор-столбцы из пространства E^n . Записав их в строку, получим матрицу

$$\mathbf{A}_m = (a_1, a_2, \dots, a_m),$$

составленную из вектор-столбцов. Причем для $m = 1$ $\mathbf{A}_1 (a_1)$, для $m = 2$ $\mathbf{A}_2 (a_1, a_2)$ и т.д. Отсюда следует общая форма записи указанной матрицы:

$$\mathbf{A}_j = (\mathbf{A}_{j-1}, a_j), j = 2, 3, \dots, m,$$

где $\mathbf{A}_1 = (a_1)$.

Поскольку мы умеем определить матрицу \mathbf{A}_1^+ , то можно построить удобный метод вычисления псевдообратных матриц, позволяющих по матрицам \mathbf{A}_1 и \mathbf{A}_1^+ построить матрицу \mathbf{A}_2^+ , затем по матрицам \mathbf{A}_2 и \mathbf{A}_2^+ — матрицу \mathbf{A}_3^+ и т.д. Как следует из сказанного, для построения метода

необходимо получить формулу, связывающую матрицы A_m^+ и A_{m+1}^+ . Такая формула следует из теоремы Гревилля, которая заключается в следующем:

если $A_{m+1} = (A_m, a_{m+1})$, то

$$A_{m+1}^+ = \begin{pmatrix} A_m^+ & (I - a_{m+1} K_m^T) \\ \dots\dots\dots & K_m^T \\ & K_{m+1}^T \end{pmatrix}, \quad (119)$$

где

$$K_{m+1} = \begin{cases} \frac{(I - A_m A_m^+) a_{m+1}}{\|(I - A_m A_m^+) a_{m+1}\|}, & \text{если } (I - A_m A_m^+) a_{m+1} \neq 0 \\ \frac{(A_m^+)^T A_m^+ a_{m+1}}{1 + \|A_m^+ a_{m+1}\|^2} & \text{— в остальных случаях.} \end{cases} \quad (120)$$

Доказывать эту теорему не будем, а сделаем лишь некоторые пояснения. Обозначение $\left(\frac{A}{K}\right)$ есть матрица A , к которой снизу дописана матрица K , что приводит к соответствующему увеличению строк в исходной матрице. Кроме того, равенство $(I - A_m A_m^+) a_{m+1} = 0$ имеет место тогда и только тогда, когда векторы a_1, a_2, \dots, a_{m+1} линейно зависимы. Иначе говоря, вычисления проводятся по верхней части формулы (120), если к матрице добавляется линейно независимый столбец, и по нижней части формулы (120), если добавляемый столбец вместе с остальными векторами образует линейно зависимую комбинацию. Вспомнив определение ранга матрицы, можно утверждать, что добавление линейно зависимого столбца к матрице не меняет ее ранга.

Существуют аналогичные формулы, позволяющие вычислять псевдообратную матрицу при добавлении к исходной матрице строки.

Вычисления по формулам (119), (120) сводятся к операциям над матрицами, которые легко программируются. Однако следует иметь в виду одно обстоятельство, затрудняющее реализацию этих формул на ЭВМ, которое связано с тем, что обычно матрицы, подлежащие псевдообращению, известны лишь приближенно. Это справедливо и в том случае, когда матрицы известны точно, поскольку при вводе их в ЭВМ и при дальнейших операциях с ними всегда имеют место ошибки округления. На стадии проверки условий формулы (120) небольшие ошибки, возникающие из-за округления или неточности измерения, могут привести к большим погрешностям в конечном результате. В такой ситуации принято говорить, что метод неустойчив к ошибкам округления. Чтобы сгладить влияние этих ошибок, можно поступить следующим образом: вместо сравнения $(I - A_m A_m^+) a_{m+1} = 0$ осуществлять проверку условия $\|(I - A_m A_m^+) a_{m+1}\| < \epsilon$, где ϵ — малое положительное число. Если это условие выполнено, то следует действовать в соответствии со второй строкой формулы (120), т.е. считать, что выполнено условие

$$(I - A_m A_m^+) a_{m+1} = 0.$$

При принятии оперативных решений часто встречается необходимость

обработки потока данных, которые образуют временную последовательность, например, результаты измерения скорости автомобиля в различные моменты времени могут образовать такую последовательность. Обычно ставится задача, чтобы эти данные обрабатывались по мере их поступления. Кроме того, желательно, чтобы объем вычислений при поступлении очередной порции данных был не очень большим, и при вычислениях на каждом шаге использовались результаты предыдущей обработки (требование рекуррентности).

Представим себе вычислительное устройство, предназначенное для экстраполяции параметров какого-либо процесса (например, погоды), данные о текущем состоянии которого поступают в дискретные моменты времени. Естественно потребовать, чтобы наше воображаемое устройство с приходом очередной порции данных корректировало экстраполированные параметры с учетом вновь поступившей информации. Поскольку речь идет об обработке временных последовательностей, то поступающая информация стареет (на состояние погоды, которая будет завтра, большее влияние оказывает сегодняшнее состояние атмосферы, нежели то состояние, которое существовало несколько месяцев назад). Поэтому алгоритм экстраполяции должен быть построен с учетом возможности исключения устаревшей информации.

Обычно для такого типа расчетов применяют метод наименьших квадратов. Напомним существо этого метода и приведем его рекуррентную форму, приспособленную для обработки временных последовательностей.

Сформулируем задачу следующим образом: экспериментатор наблюдает значения неизвестной функции в дискретные моменты времени; требуется описать поведение этой функции в те моменты времени, когда эта функция не наблюдалась, например, между отдельными наблюдениями или даже вне интервала наблюдения.

Для решения этой задачи выбирают некоторое конечное семейство функций времени $f_1(t), f_2(t), \dots, f_m(t)$. Чаще всего эти функции представляют собой следующий набор:

$$f_1(t) = 1, f_2(t) = t, f_3(t) = t^2, \dots, f_m(t) = t^{m-1}.$$

Далее рассматривают множество функций, являющееся множеством линейных комбинаций этого набора, т.е. множеством функций вида $\sum_{i=1, m} \xi_i f_i(t)$, где ξ_i — произвольные числа. И наконец, из этого множества функций выбирают такую, которая оказывается наиболее близкой к неизвестной функции в точках наблюдения. На понятии близости функций остановимся подробнее. Предварительно заметим, что любая из функций, входящая в упомянутое выше множество линейных комбинаций исходных функций, также является функцией времени.

Пусть значение неизвестной функции наблюдается в различные моменты времени, т.е.

$$z_i = f(t_i), i = \overline{1, n}.$$

Введем вектор-столбец $Z = (z_1, \dots, z_n)^T$.

Значения функций $f_j(t)$ из имеющегося набора в те же моменты времени обозначим так:

$$h_{ij} = f_j(t_i), i = \overline{1, n}, j = \overline{1, m}.$$

Получим набор из $m \times n$ чисел, который можно записать в виде матрицы:

$$H = \begin{pmatrix} h_{11} & h_{12} & \dots & h_{1m} \\ h_{21} & h_{22} & \dots & h_{2m} \\ \dots & \dots & \dots & \dots \\ h_{n1} & h_{n2} & \dots & h_{nm} \end{pmatrix} = \begin{pmatrix} f_1(t_1) f_2(t_1) \dots f_m(t_1) \\ f_1(t_2) f_2(t_2) \dots f_m(t_2) \\ \dots & \dots & \dots & \dots \\ f_1(t_n) f_2(t_n) \dots f_m(t_n) \end{pmatrix}.$$

Введем еще один вектор-столбец, составленный из коэффициентов линейной комбинации набора $f_j(t)$, $j = \overline{1, m}$:

$$X = (\xi_1, \dots, \xi_m)^T.$$

Умножим матрицу H на вектор X :

$$\begin{pmatrix} h_{11} & h_{12} & \dots & h_{1m} \\ h_{21} & h_{22} & \dots & h_{2m} \\ \dots & \dots & \dots & \dots \\ h_{n1} & h_{n2} & \dots & h_{nm} \end{pmatrix} \begin{pmatrix} \xi_1 \\ \xi_2 \\ \vdots \\ \xi_m \end{pmatrix} = \begin{pmatrix} \xi_1 h_{11} + \xi_2 h_{12} + \dots + \xi_m h_{1m} \\ \xi_1 h_{21} + \xi_2 h_{22} + \dots + \xi_m h_{2m} \\ \dots & \dots & \dots & \dots \\ \xi_1 h_{n1} + \xi_2 h_{n2} + \dots + \xi_m h_{nm} \end{pmatrix}$$

Из анализа полученного результата видно, что при умножении матрицы H на вектор X получается вектор, являющийся линейной комбинацией столбцов матрицы H . Полученный вектор представляет собой значения некоторой функции из заданного набора в моменты времени t_i , $i = \overline{1, n}$.

Рассмотрим следующую величину:

$$\|Z - HX\|^2.$$

Эта величина представляет собой расстояние между вектором Z — набором имеющихся наблюдений, и вектором, представляющим собой линейную комбинацию значений заданного набора функций в те же моменты времени. Поскольку выбор коэффициентов линейной комбинации (выбор вектора X) находится в нашей власти, то его, естественно, следует осуществить так, чтобы введенное расстояние было минимальным.

Как было показано выше, одним из решений поставленной задачи является вектор, определяемый по формуле

$$X = H^+ Z, \tag{121}$$

где H^+ — матрица, псевдообратная к матрице H .

Найдя значения весовых коэффициентов по формуле (121), можно представить оценку неизвестной функции в виде $h^T(t) X$, где $h^T(t) = [f_1(t), f_2(t), \dots, f_m(t)]$.

Полученная оценка оказывается наиболее близкой в указанном выше смысле к неизвестной функции в наблюдаемые моменты времени

$t_i, i = 1, \bar{n}$, что позволяет надеяться, что и в другие моменты времени это отклонение не будет уж столь велико.

Из изложенного ясно, что решение задачи минимизации величины $\|Z - HX\|^2$ фактически является методом наименьших квадратов, который часто применяется для сглаживания, фильтрации и прогнозирования.

Рассмотрим рекуррентную форму вычислений по этому методу.

Пусть $Z_n (n = 1, 2, \dots)$ — последовательность поступающих наблюдений, H_n — матрица, задающая значения имеющихся функций X_n — набор весовых коэффициентов, минимизирующих отклонение $\|Z_n - H_n X_n\|^2$. Ранее было показано, что $X_n = H_n^+ Z_n$.

Пусть поступило $(n + 1)$ -е значение наблюдения ξ_{n+1} . Тогда вектор наблюдений будет иметь следующий вид:

$$Z_{n+1} = \begin{pmatrix} Z_n \\ \dots \\ \xi_{n+1} \end{pmatrix} \in L_{n+1, 1}.$$

Соответствующая матрица значений заданного набора функций будет выглядеть так:

$$H_{n+1} = \begin{pmatrix} H_n \\ \dots \\ h_{n+1}^T \end{pmatrix} \in L_{n+1, m}.$$

Размерность вектора весовых коэффициентов останется прежней, т.е. $X_{n+1} \in L_{m, 1}$, но его значение изменится и будет вычисляться по формуле

$$X_{n+1} = H_{n+1}^+ \cdot Z_{n+1}. \quad (122)$$

Раскроем формулу (122) в соответствии с теоремой Гревилля. Тогда получим

$$\begin{aligned} X_{n+1} &= H_{n+1}^+ Z_{n+1} = [(I - K_{n+1} h_{n+1}^T) H_n^+, K_{n+1}] \begin{pmatrix} Z_n \\ \dots \\ \xi_{n+1} \end{pmatrix} = \\ &= (I - K_{n+1} h_{n+1}^T) H_n^+ Z_n + K_{n+1} \xi_{n+1} = X_n - K_{n+1} h_{n+1}^T X_n + \\ &+ K_{n+1} \xi_{n+1} = X_n + K_{n+1} (\xi_{n+1} - h_{n+1}^T X_n). \end{aligned} \quad (123)$$

Заметим, что величина $h_{n+1}^T X_n$ есть экстраполированное значение в момент времени t_{n+1} , и этот прогноз сделан на основе первых n наблюдений. Действительно,

$$\begin{aligned} h_{n+1}^T X_n &= [f_1(t_{n+1}), f_2(t_{n+1}), \dots, f_m(t_{n+1})] (\xi_1^n, \xi_2^n, \dots, \xi_m^n)^T = \\ &= f_1(t_{n+1}) \xi_1^n + f_2(t_{n+1}) \xi_2^n + \dots + f_m(t_{n+1}) \xi_m^n, \end{aligned}$$

т.е. рассматриваемая величина есть линейная комбинация значений известного набора функций в моменты времени t_{n+1} , которая и является оценкой значения ξ_{n+1} , сделанной по n наблюдениям.

Из анализа формулы (123) следует, что новое значение весовых коэффициентов равно старому значению, полученному по наблюдениям,

плюс значения поправки, равной разности между экстраполированным значением и истинным наблюдением, умноженной на некоторый коэффициент.

Таким образом, получена новая рекуррентная форма МЛЭ, позволяющая решать задачи экстраполяции даже в том случае, когда некоторые из векторов, образующих матрицу \mathbf{H} , оказываются линейно зависимыми. Эта форма задается следующими уравнениями:

$$\mathbf{X}_{n+1} = \mathbf{X}_n + \mathbf{K}_{n+1} (\xi_{n+1} - \mathbf{h}_{n+1}^T \mathbf{X}_n), \quad (124)$$

где

$$\mathbf{K}_{n+1} = \begin{cases} \frac{(\mathbf{I} - \mathbf{H}_n^+ \mathbf{H}_n) \mathbf{h}_{n+1}}{\|\mathbf{I} - \mathbf{H}_n^+ \mathbf{H}_n\| \mathbf{h}_{n+1}}, & \text{если } (\mathbf{I} - \mathbf{H}_n^+ \mathbf{H}_n) \mathbf{h}_{n+1} \neq 0; \\ \frac{\mathbf{H}_n^+ (\mathbf{H}_n^T)^+ \mathbf{h}_{n+1}}{1 + \mathbf{h}_{n+1}^T \mathbf{H}_n^+ (\mathbf{H}_n^T)^+ \mathbf{h}_{n+1}}, & \text{если } (\mathbf{I} - \mathbf{H}_n^+ \mathbf{H}_n) \mathbf{h}_{n+1} = 0. \end{cases} \quad (125)$$

2. УЧЕТ СТАРЕНИЯ ИСХОДНОЙ ИНФОРМАЦИИ В МЛЭ

Полученные ранее формулы Гревия позволяют вычислять псевдообратные матрицы по мере появления очередных строк или столбцов в исходной матрице. Рассмотрим теперь такой вопрос: как ведет себя псевдообратная матрица, если из нее исключаются некоторые строки?

Чтобы уяснить необходимость исключения строк из матрицы, представим себе следующую ситуацию: наблюдаются перемещения некоторого объекта, причем информация о его состоянии поступает в дискретные моменты времени и зашумлена. Поскольку информация зашумлена, то возникает необходимость накопления достаточного ее количества, чтобы оценить состояние наблюдаемого объекта на фоне шума. Но с другой стороны, наблюдения за движущимся объектом образуют временную последовательность, что приводит к старению информации и необходимости ее исключения.

Для решения поставленной задачи вначале уясним себе, как меняется псевдообратная матрица при перестановке строк или столбцов в исходной матрице. Операцию перестановки строк (для определенности будем выполнять именно эту операцию, но все рассуждения будут справедливы для столбцов) можно представить как умножение исходной матрицы на матрицу специального вида.

Возьмем единичную матрицу

$$\mathbf{I} = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 1 \end{pmatrix} \in L_{n,n}$$

Переставим в ней произвольным образом строки. Матрицу, полученную таким образом, обозначим буквой \mathbf{P} . Например, для $n = 3$

$$I = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Поменяв местами первую и последнюю строки, получим следующую матрицу:

$$P = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

Возьмем произвольную матрицу $A \in L_{n, k}$ и найдем произведение P на A . Для упрощения выкладок примем $n = 3, k = 2$. В этом случае матрица будет иметь вид

$$A = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \end{pmatrix}.$$

Тогда

$$P \cdot A = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \end{pmatrix} = \begin{pmatrix} a_{31} & a_{32} \\ a_{21} & a_{22} \\ a_{11} & a_{12} \end{pmatrix},$$

т.е. в результате умножения получаем матрицу, в которой переставлены первая и третья строки, т.е. произведена та же операция, которая привела к образованию матрицы P из единичной матрицы.

После размышления над "механикой" умножения матриц нетрудно понять, что этот факт справедлив не только для рассмотренного примера. В общем случае справедливо следующее утверждение: если в единичной матрице переставить строки и умножить полученную матрицу $P \in L_{n, n}$ на произвольную матрицу $A \in L_{n, k}$, то результат будет отличаться от заданной произвольной матрицы лишь перестановкой строк, причем эта перестановка совпадает с перестановкой, произведенной в единичной матрице. Тем самым операция перестановки строк в исходной матрице сведена к операции умножения слева на матрицу специального вида. Аналогичные рассуждения показывают, что для перестановки столбцов произвольной матрицы $A \in L_{n, k}$ ее можно умножить на матрицу $P \in L_{k, k}$, которая получена из единичной матрицы необходимой перестановкой столбцов.

Матрицы, полученные перестановкой строк (столбцов) из единичной матрицы, имеют еще одно положительное свойство: для их обращения (а значит, и псевдообращения) достаточно их транспонировать, т.е. имеет место тождество

$$P \cdot P^T = P^T \cdot P = I. \quad (126)$$

Из (126) следует, что

$$(PA)^+ = A^+ P^+ = A^+ P^T. \quad (127)$$

Подробное обоснование формулы (127) дано в работе [3]. Отсюда следует правило: чтобы псевдообратить матрицу A , в которой строки переставлены в результате ее умножения на матрицу P , достаточно в матрице A^+ аналогично переставить столбцы. Другими словами, перестановка строк (столбцов) в исходной матрице ведет к такой же перестановке столбцов (строк) в псевдообратной матрице.

Запомним это правило, вернемся к задаче вычеркивания строк из матрицы. Вначале рассмотрим пример исключения из исходной матрицы последней строки. Пусть

$$H_{n+1} = \begin{pmatrix} H_n \\ \dots \\ h_{n+1}^T \end{pmatrix}.$$

Теорема Гревилля позволяет по матрицам H_n и H_n^+ найти матрицу H_{n+1}^+ , а именно:

$$H_{n+1}^+ = [(I - K_{n+1} h_{n+1}^T) H_n^+, K_{n+1}], \quad (128)$$

где K_{n+1} определяется по формуле (125).

Обозначив

$$G_n = (I - K_{n+1} h_{n+1}^T) H_n^+,$$

представим уравнение (128) в виде

$$H_{n+1}^+ = (G_n, K_{n+1}),$$

где K_{n+1} — вектор-столбец.

Таким образом, схема вычисления псевдообратной матрицы при добавлении к исходной матрице $(n+1)$ -й строки такова: вначале по матрицам H_n^+ , H_n , h_{n+1} вычисляется $(n+1)$ -й столбец матрицы H_{n+1}^+ , а затем матрица $G_n = (I - K_{n+1} h_{n+1}^T) H_n^+$, представляющая собой n первых столбцов матрицы H_{n+1}^+ .

Предположим, что нам известны матрицы

$$H_{n+1} = \begin{pmatrix} H_n \\ \dots \\ h_{n+1}^T \end{pmatrix}, H_{n+1}^+ = (G_n, K_{n+1}).$$

Вычеркивая последнюю строку из матрицы H_{n+1} , превращаем ее в матрицу H_n . Какую операцию нужно выполнить, чтобы превратить матрицу H_{n+1}^+ в матрицу H_n^+ ? Оказывается для этого нужно вычеркнуть из матрицы H_{n+1}^+ последний столбец, в результате чего получится матрица G_n . Далее матрицу H_n^+ вычисляют по следующей формуле:

$$H_n^+ = (I - K_{n+1} h_{n+1}^T)^+ \cdot G_n, \quad (129)$$

где

$$(I - K_{n+1} h_{n+1}^T)^+ = \begin{cases} I - \frac{K_{n+1} h_{n+1}^T + 1}{1 - h_{n+1}^T K_{n+1}}, & \text{если } h_{n+1}^T K_{n+1} \neq 1; \\ I + \frac{K_{n+1} h_{n+1}^T + 1}{\|K_{n+1}\|^2 \|h_{n+1}\|^2} - \frac{K_{n+1} K_{n+1}^T}{\|K_{n+1}\|^2} - \\ - \frac{h_{n+1}^T h_{n+1}}{\|h_{n+1}\|^2}, & \text{если } h_{n+1}^T K_{n+1} = 1. \end{cases}$$

В справедливости этих формул можно убедиться, проверив выполнение условий Пенроуза для матриц $(I - K_{n+1} h_{n+1}^T)$, $(I - K_{n+1} \times h_{n+1}^T)^+$.

Все сказанное выше относилось к вычеркиванию из матрицы последней строки. Покажем, что к этому случаю можно свести и более общий случай добавления или вычеркивания произвольной строки.

Пусть к исходной матрице добавлена произвольная строка. Переставим эту строку на последнее место, т.е. образуем последнюю строку. Как было уже показано, перестановка строк в исходной матрице приводит к такой же перестановке столбцов в псевдообратной матрице. Последнюю матрицу можно рассчитать по приведенным выше формулам и, осуществив обратную перестановку столбцов согласно тому же правилу, получить псевдообратную матрицу, в которую была добавлена строка.

Дадим математическую интерпретацию сказанному. Пусть в матрицу

$$H_n = \begin{pmatrix} h_1^T \\ h_2^T \\ \vdots \\ h_n^T \end{pmatrix}$$

после строки с номером i добавлена строка h_{i+1}^T . Тогда для вычисления псевдообратной матрицы нужно выполнить следующие действия.

1. Вычислить вектор-столбец

$$K = \begin{cases} \frac{(I - H_n^+ H_n) h_{i+1}}{\|(I - H_n^+ H_n) h_{i+1}\|^2}, & \text{если } (I - H_n^+ H_n) h_{i+1} \neq 0; \\ \frac{H_n^+ (H_n^T)^+ h_{i+1}}{1 + h_{i+1}^T H_n^+ (H_n^T)^+ h_{i+1}}, & \text{если } (I - H_n^+ H_n) h_{i+1} = 0. \end{cases}$$

2. Вычислить матрицу

$$G = (I - K h_i^T) H_n^+.$$

3. Вставить в матрицу G столбец K после i -го столбца.

Аналогично рассуждая, можно получить алгоритм удаления из матрицы произвольной строки. Схема вычислений псевдообратной матрицы в этом случае следующая. Пусть из матрицы

$$H_{n+1} = \begin{pmatrix} h_1^T \\ h_2^T \\ \vdots \\ h_{n+1}^T \end{pmatrix}$$

вычеркнута строка h_i^T , $1 \leq i < n+1$. Обозначим оставшуюся матрицу через H_n . Тогда для вычислений H_n^+ нужно выполнить следующие действия:

- 1) из матрицы H_{n+1}^+ вычеркнуть i -й столбец;
- 2) оставшуюся матрицу умножить слева на матрицу $(I - K h_i^T)^+$, где K — столбец, вычеркнутый из матрицы H_{n+1}^+ ;
- 3) матрицу $(I - K h_i^T)^+$ рассчитать по формуле

$$(I - K h_i^T)^T = \begin{cases} I + \frac{K h_i^T}{1 - h_i^T K}, & \text{если } h_i^T K \neq 1; \\ I + \frac{K h_i^T}{\|K\|^2 \|h_i\|^2} - \frac{h_i h_i^T}{\|h_i\|^2} - \\ - \frac{K^T K}{\|K\|^2}, & \text{если } h_i^T K = 1. \end{cases}$$

Программа, реализующая рекуррентную форму метода многомерной линейной экстраполяции, была написана на ФОРТРАНЕ и строилась на следующей методологической основе: имеется некоторый набор векторов ситуаций и соответствующих им векторов решений. Из этих наборов векторов формируются две матрицы: матрица ситуаций H_{mn} и матрица решений E_{mn} , где m — число векторов в наборе; n — число точек наблюдения. Далее предполагается, что имеется новая ситуация Z_{n1} , отличная от имеющихся в наборе, но в некотором смысле близкая к ним, где n_1 — число точек наблюдения новой ситуации (обычно $n_1 \leq n$). Рассматриваемая программа позволяет найти новое решение, соответствующее ситуации Z_{n1} .

Рассмотрим упрощенную структурную схему программы (рис. 34).

В блоке наращивания информации происходит постепенное заполнение матрицы ситуаций и корректировка псевдообратной матрицы. Если есть необходимость, а эта необходимость возникает при обработке временных последовательностей, то в блоке исключения информации происходит постепенное (на каждом шаге) отбрасывание устаревшей информации, корректировка матрицы ситуаций и ее псевдообратной матрицы. Работа программ продолжается до тех пор, пока поступают параметры вектора наблюдений Z_{n1} .

В блоке определения коэффициента веса проводится минимизация величины $\|Z - HX\|^2$.

Полученные коэффициенты $X = H^+ Z$ используются в следующем

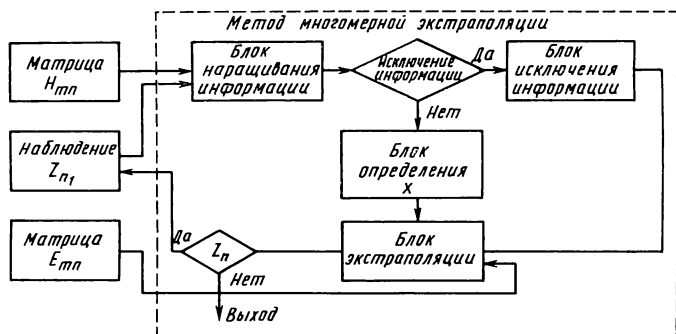


Рис. 34. Структурная схема рекуррентного алгоритма многомерной экстраполяции

Рис. 35. Восстановление процесса методом многомерной экстраполяции: 1 — точное значение процесса; 2 — прогнозируемое значение процесса

блоке — блоке экстраполяции — для определения нового решения:

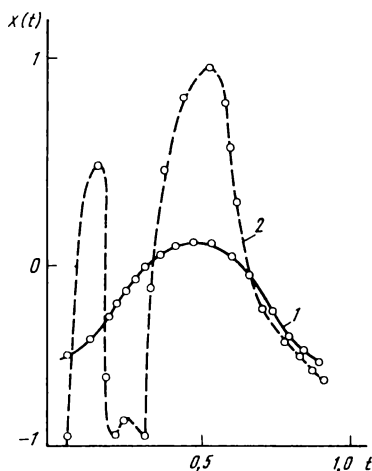
$$\theta_n = E H^+ Z_{n1}.$$

Следует заметить, что формирование матрицы ситуаций H_{mn} , матрицы решений E_{mn} и вектора новой ситуации Z_{n1} , осуществляется за пределами программы, т.е. находится в распоряжении исследователя.

Данная программа позволяет решать следующие задачи: восстановление неизвестных значений каких-либо функций, восстановление лиц, поиск экстраполированных параметров движущегося объекта и т.д.

Результаты восстановления заданного процесса с помощью алгоритма многомерной экстраполяции представлены на рис. 35. Задача решалась при следующих исходных данных: наблюдаемый процесс — одномерный; для формирования исходной информации использовались пять реализаций процесса и 20 точек наблюдения. Это значит, что матрицы и вектор наблюдений имели следующие размерности: $H (5, 20)$; $E (5, 10)$; $Z (20)$. Решение задачи прогноза осуществлялось на каждом шаге по мере заполнения вектора Z_{n1} и матрицы H_{mn} ; матрица E_{mn} была заполнена и в процессе решения не менялась. На рис. 35 кривая 1 — точное значение процесса, а кривая 2 — значение процесса, полученное с помощью алгоритма многомерной экстраполяции. Анализ полученных результатов показывает, что по мере накопления информации прогнозируемое значение процесса, полученное методом многомерной экстраполяции, сходится к точному значению.

Итак, в данной главе рассмотрено несколько направлений повышения точности метода многомерной линейной экстраполяции, позволяющих значительно расширить диапазон применения этого метода для решения задач, связанных с принятием оперативных решений на стадии предварительного проектирования.



Заключение

Основное внимание в книге уделено технике применения метода многомерной линейной экстраполяции для решения разнообразных задач проектирования и управления. Несколько меньше акцентированы вопросы теоретического плана, направленные на разработку новых модификаций метода, позволяющих решать задачи не только в детерминированной, но и в стохастической постановке. Хотелось бы обратить внимание читателя на ряд проблем (заранее оговоримся, что перечень проблем далеко не полный), решение которых позволяет определить новые грани метода многомерной линейной экстраполяции и расширить диапазон его применения.

Первая группа проблем связана с исследованием алгоритмов эвристического прогнозирования. В настоящее время интенсивно ведутся работы по совершенствованию этого метода; причем внимание исследователей все чаще концентрируется на не "внешнем формировании" указанного метода — подборе соответствующих по квалификации и числу экспертов, создании удобств для проведения оценок, разработке методов оценки результатов опроса [41], а на его внутренней структуре. Во введении уже говорилось о том, что способность предвосхищать будущее является свойством нашего сознания, качеством естественного интеллекта. Именно способность проникать в глубину изучаемого явления, способность предвидеть будущее, способность по нечетким аналогиям выбрать адекватную реальной ситуации модель отличают выдающегося ученого от остальной группы исследователей. По мнению авторов, алгоритмы, построенные с помощью метода многомерной линейной экстраполяции, наиболее близки к тем алгоритмам, которыми "пользуется" естественный интеллект. Использование подобных алгоритмов, основанных на рациональном сочетании искусственного и естественного интеллектов, может лечь в основу создания высокоэффективных систем, предназначенных для изучения новых явлений (процессов). Кроме того, изучение внутренней структуры метода эвристического прогнозирования позволит разработать более эффективные методы комбинированного прогнозирования [41], поскольку в этом случае будет создана единая математическая (методологическая) основа обработки результатов, полученных различными методами.

Вторая группа проблем касается существа метода многомерной линейной экстраполяции. И здесь в первую очередь возникают весьма важные вопросы, связанные с выбором обучающих последовательностей (задача таксономии). Многочисленные примеры, приведенные в данной книге, показывают, что "удачный" выбор обучений во многом определяет качество экстраполяции. Следовательно, решение задачи таксономии, заключающейся в разбиении множества обучений на некоторое число подмножеств (таксонов) по заданному критерию качества, приведет к

повышению точности экстраполяции как в детерминированных, так и в стохастических условиях (речь идет об условиях, когда на одну или несколько обучающих последовательностей наложен шум).

Не менее интересные вопросы связаны с повышением качества работы фильтров (в частности, фильтра Калмана) в переходном режиме. Результаты, приведенные в п. 7 гл. 3, показывают, что использование метода многомерной экстраполяции для выбора модели наблюдаемого сигнала в значительной степени повышает точность работы фильтра Калмана в переходном режиме, т.е. когда модель сигнала меняет свои параметры. Дальнейшие исследования в этом направлении могут привести к созданию адаптивного фильтра Калмана (адаптивного с точки зрения модели наблюдаемого сигнала).

Говоря о повышении точности метода многомерной экстраполяции и расширении диапазона его применения, нельзя не коснуться направлений исследования, связанных с использованием робастных оценок, нечетных (лингвистических) переменных и т.д.

Итак, метод многомерной экстраполяции находится в развитии. Результаты, приведенные в книге, характеризуют определенный этап этого развития. Исследования в указанных выше направлениях позволят, с одной стороны, повысить качество экстраполяции, а с другой — расширить диапазон применения этого метода для решения задач, рассмотрение которых осталось за пределами этой книги.

СПИСОК ЛИТЕРАТУРЫ

1. Агладзе В.А., Пономарев Ю.П. Рекуррентная форма метода многомерной линейной экстраполяции. — В кн.: Адаптация в вычислительных системах. Рига: Зинатне, 1978, с. 77–84.
2. Агладзе В.А., Пономарев Ю.П. Рекуррентная форма метода многомерной линейной экстраполяции. — В кн.: Машинные методы обнаружения закономерностей. Рига: Рижский политехн. ин-т, 1981, с. 73 – 76.
3. Алберт А. Регрессия, псевдоинверсия и рекуррентное оценивание. М.: Наука, 1977. 22 с.
4. Брайсон А., Хо Ю-Ши. Прикладная теория оптимального управления. М.: Мир, 1972. 482 с.
5. Вапник В.Н., Червоненкис А.Я. Теория распознавания образов. М.: Наука, 1974. 416 с.
6. Гаврилко Б.П., Загоруйко Н.Г. Универсальный алгоритм эмпирического предсказания. — В кн.: Вычислительные системы. Новосибирск: Ин-т матем., 1973, вып. 55, с. 134 – 138.
7. Дуда Р., Харт П. Распознавание образов и анализ сцен. М.: Мир, 1976. 511 с.
8. Декснис Г.К., Растригин Л.А. Синтез алгоритмов поисковой оптимизации методом многомерной линейной экстраполяции. — В кн.: Проблемы случайного поиска. Рига: Зинатне, 1978, с. 184–197.
9. Зайдель С.М., Кирпичев Ю.П., Растригин Л.А. Повышение точности метода многомерной экстраполяции при оптимальном проектировании. — Автоматика и вычислительная техника, № 1, 1970, с. 3 – 8.
10. Заде Л. Основы нового подхода к анализу сложных систем и процессов принятия решения. — В кн.: Математика сегодня. М.: Знание, 1974, с. 5 – 48.
11. Загоруйко Н.Г. Методы распознавания образов и их применение. М.: Сов. радио, 1972. 208 с.
12. Ивахненко А.Г., Лапа В.Г. Предсказание случайных процессов. Киев: Наукো-ва думка, 1971. 416 с.
13. Ивахненко А.Г. Долгосрочное прогнозирование и управление сложными системами. Киев: Техніка, 1975. 310 с.
14. Капустян В.М., Махотенко Ю.А. "Белые пятна" приборостроения. — Приборы и системы управления, 1977, № 9, с. 24 – 25.
15. Катковник В.Я. Линейные и нелинейные методы непараметрического регрессионного анализа. — Автоматика, 1979, № 5, с. 35 – 46.
16. Клаузевиз К. О войне. М.: Воениздат, 1936. 231 с.
17. Колмогоров А.Н. Автоматы и жизнь. — В кн.: Кибернетика ожидаемая и кибернетика неожиданная. М.: Наука, 1986, с. 12 – 31.
18. Красненкер А.С. Задачи и методы векторной оптимизации. — Измерение, контроль, автоматизация, 1975, № 1, с. 51 – 56.
19. Лейтман Дж. Введение в теорию оптимального управления. М.: Наука, 1968. 272 с.
20. Мелешко В.И. Устойчивое к возмущениям псевдообращение замкнутых операторов. — Журнал вычислительной математики и математической физики. Т. 17, 1977, № 15, с. 78 – 84.
21. Поспелов Г.С., Баришполец В.А. Интеграция науки и производства. М.: Знание, 1978. 64 с.
22. Пономарев Ю.П., Агладзе В.А., Гавырина Л.В. Решение задач методом многомерной экстраполяции. — В кн.: Адаптация в вычислительных системах. Рига: Зинатне 1978, с. 129 – 138.
23. Поцелуев А.В. Статистический анализ и синтез сложных динамических систем. М.: Машиностроение. 1984. 208 с.

24. Пономарев Ю.П., Растригин Л.А. Некоторые вопросы обнаружения закономерностей. — В кн.: Кибернетика и вычислительная техника. Киев: Наукова думка, 1979, вып. 46, с. 36 — 41.
25. Программная реализация метода многомерной экстраполяции /Ю.П. Пономарев, В.А. Агладзе, Л.В. Гавырина, Г.А. Агладзе. — В кн.: Машинные методы обнаружения закономерностей. Рига: Рижский политехн. ин-т, 1981, с. 169 — 174.
26. Растригин Л.А. Проблема обнаружения закономерностей. — В кн.: Машинные методы обнаружения закономерностей. Рига: Рижский политехн. ин-т, 1981, с. 5 — 9.
27. Растригин Л.А., Трахтенберг В.С. Применение экстраполяции при оптимальном проектировании сложных систем. — В кн.: Методы статистической оптимизации. Рига: Зинатне, 1968, с. 43 — 58.
28. Растригин Л.А., Трахтенберг В.С. Оценка числовых характеристик функций методом многомерной экстраполяции. — В кн.: Проблемы статистической оптимизации. Рига: Зинатне, 1968, с. 179 — 189.
29. Растригин Л.А. Адаптация сложных систем. Рига: Зинатне, 1981. 375 с.
30. Растригин Л.А., Фабрикант М.И. Использование методов экстраполяции для решения задач распознавания образов. — В кн.: Распознавание образов. Рига: Зинатне, 1974, с. 4 — 12.
31. Растригин Л.А., Фабрикант М.И., Эренштейн Р.Х. Два локальных подхода к решению задач медицинской диагностики. — В кн.: Кибернетика и вычислительная техника. Киев: Наукова думка, 1976, вып. 33 с. 57 — 62.
32. Растригин Л.А. Статистические методы поиска. М.: Наука, 1968. 376 с.
33. Растригин Л.А., Фабрикант М.И. Локальные экстраполяционные модели сложных объектов управления. — В кн.: Управление сложными системами. Вып. 4. Рига: Рижский политехн. ин-т, 1976, с. 65 — 72.
34. Растригин Л.А., Фабрикант М.И. Эмпирическое прогнозирование методом многомерной линейной экстраполяции. — В кн.: Машинные методы обнаружения закономерностей. Новосибирск: Ин-т матем., 1976, с. 85 — 89.
35. Растригин Л.А., Самсонс Д.В. Метод многомерной линейной экстраполяции в задачах выявления динамических закономерностей. — В кн.: Машинные методы обнаружения закономерностей. Рига: Рижский политехн. ин-т, 1981, с. 68 — 73.
36. Растригин Л.А., Эйдук Я.Ю. Поисковые алгоритмы определения множества Парето. — В кн.: Адаптация в вычислительных системах. Рига: Зинатне, 1978, с. 69 — 76.
37. Растригин Л.А. Системы экстремального управления. М.: Наука, 1974. 630 с.
38. Стечкин С.Б., Субботин Ю.Н. Сплайны в вычислительной математике. М.: Наука, 1976. 248 с.
39. Сударев Ю.Н. Понятие пространства в математике. — Знание, сер. Математика, кибернетика, 1974, № 12. 64 с.
40. Экстраполяционные модели оперативного управления режимами электроэнергетических систем /А.Ф. Аб, Ю.П. Первушин, Л.А. Растригин, М.И. Фабрикант. — Известия АН СССР. Энергетика и транспорт, 1977, с. 3 — 13.
41. Чуев Ю.В., Михайлов Ю.Б., Кузьмин В.И. Прогнозирование количественных характеристик. М.: Сов. радио, 1975. 398 с.

ОГЛАВЛЕНИЕ

Предисловие	3
Введение	4
Глава I. Теоретические основы построения алгоритма принятия оперативных решений	9
1. Математическое прогнозирование	9
2. Экстраполяция в проектировании и управлении	11
3. Требования, предъявляемые к методу экстраполяции	13
4. Экстраполяция как синтез модели (обзор основных методов)	14
5. Линейные пространства и линейные преобразования	20
6. Алгоритм многомерной линейной экстраполяции с евклидовой метрикой	29
7. Анализ погрешностей метода многомерной экстраполяции при восстановлении линейных и нелинейных операторов систем	33
8. Алгоритм многомерной линейной экстраполяции на подвижных подпространствах ситуаций и решений	40
Глава II. Применение алгоритма многомерной экстраполяции для решения задач проектирования	44
1. Восстановление неизвестных характеристик проектируемого объекта	44
2. Восстановление числовых таблиц	49
3. Оценка числовых характеристик функции	55
4. Использование метода экстраполяции для решения задач распознавания образов	61
Глава III. Применение алгоритма многомерной экстраполяции для решения задач управления	66
1. Метод многомерной экстраполяции и задачи управления динамической системой	66
2. Экстраполяционные модели оперативного управления режимами электроэнергетических систем	72
3. Синтез алгоритмов поисковой оптимизации методом МЛЭ	79
4. Эмпирическое прогнозирование методом многомерной линейной экстраполяции	85
5. Метод многомерной линейной экстраполяции в задачах выявления динамических закономерностей	89
6. Поисковые алгоритмы определения множества Парето	91
7. Фильтр Калмана и метод многомерной экстраполяции	95
Глава IV. Пути совершенствования алгоритма многомерной экстраполяции	99
1. Рекуррентная форма МЛЭ	99
2. Учет старения исходной информации в МЛЭ	106
Заключение	112
Список литературы	114

Производственное издание

Леонид Андреевич Растргин
Юрий Петрович Пономарев

**ЭКСТРАПОЛЯЦИОННЫЕ МЕТОДЫ
ПРОЕКТИРОВАНИЯ И УПРАВЛЕНИЯ**

Редактор *А.В. Лысенко*
Художественный редактор *С.С. Водчиц*
Обложка художника *В.Д. Епанешникова*
Технический редактор *Н.В. Павлова*
Корректор *Л.В. Тарасова*

ИБ № 3672

Сдано в набор 6.09.85.	Подписано в печать 10.04.86.	Т- 09129.
Формат 60×90 1/16.	Бумага офсетная № 2.	Гарнитура Пресс Роман.
Печать офсетная.	Усл.печл. 7,5.	Усл.кр.-отт. 7,75.
Тираж 5100.	Заказ 376.	Уч.-издл. 8,25.
		Цена 40 к.

Ордена Трудового Красного Знамени издательство "Машиностроение",
107076, Москва, Стромынский пер., 4.

Отпечатано в Московской типографии № 8 Союзполиграфпрома
при Государственном комитете СССР по делам издательств, полиграфии
и книжной торговли 101898, Москва, Хохловский пер., 7, с оригинал-макета,
изготовленного в издательстве "Машиностроение" на наборно-пишущих машинах

**Книги для специалистов,
работающих в области автоматики и телемеханики
выпустит
издательство "Машиностроение"
в 1987 году**

ПРОИЗВОДСТВЕННАЯ ЛИТЕРАТУРА

А х м е т ш и н Н.И., Г о ц Э.М., Р о д и к о в Н.Ф. Вибрационное резание металлов/ Под ред. К.М. Рагульскаса. — 6 л.: ил. — (Б-ка инженера, Вибрационная техника). — (В обл.): 40 к.

Рассмотрены вопросы использования инерционно-импульсных приводов в качестве возбудителей колебаний инструмента или обрабатываемой детали для вибрационного резания металлов. Дана классификация способов дробления стружки и способов вибрационного резания, обоснована перспективность их применения.

Приведены структурные и кинематические схемы приводов, результаты экспериментальных исследований инерционно-импульсных приводов при резании с различными видами вибраций.

Для инженерно-технических работников машиностроительных заводов и научно-исследовательских институтов.

Гибкие производственные системы Японии: Пер. с яп. — 20 л.: ил. — Токио, Япония, 1983. — (В пер.): 1 р. 70 к.

Сборник примеров 60 действующих японских гибких производственных систем (ГПС) отражает значительный опыт Японии в их создании и использовании. Книга отличается большим статистическим материалом по эффективности внедрения ГПС, построению программного обеспечения, сравнительным данным ГПС разных стран.

Заслуживают внимания разработки новой системы конкретного предприятия и анализа ее эффективности.

Для инженеров, занимающихся вопросами автоматизации производства.

Е ф р е м о в а Т.К., Тагаевская А.А., Шубин А.Н. Пневматические комплексы технических средств автоматизации. — 20 л.: ил. — (В пер.): 1 р. 40 к.

Рассмотрены комплексы пневматических средств "Центр", "Режим-1", "Цикл", "Компас" и "Центр — Логика", предназначенные для автоматизации непрерывных и периодических технологических процессов. По каждому комплексу даны: структура, реализуемые функции, набор приборов и субблоков, методика проектирования систем автоматического управления или синтез дискретных управляющих устройств, примеры автоматизации с использованием данного комплекса.

Для инженерно-технических работников, занимающихся созданием и эксплуатацией пневматических систем автоматического управления технологическими процессами в различных отраслях промышленности.

Контактные системы / В.В. Осташевичус, Б.В. Рудгальвис, В.Л. Рагульскене, Б.П. Бакшис. — 18 л.: ил. — (В пер.): 1 р. 20 к.

Рассмотрены конструкции и области применения контактных систем (КС), основные динамические критерии, оценивающие работу этих систем, методы инженерных расчетов и проектирования. Изложены оригинальные методы и средства измерений параметров движения и упругих деформаций и на их базе разработанные методы и средства контроля, диагностики и прогнозирования работы КС.

Для инженерно-технических работников, занимающихся разработкой, изготовлением и эксплуатацией КС в прецизионном машиностроении и приборостроении.

Конюхов Н.Е., Медников Ф.М., Нечаевский М.Л. Электромагнитные датчики механических величин. — 20 л.: ил. — (В пер.): 1 р. 40 к.

Изложены вопросы теории, расчета и проектирования датчиков перемещения, ускорения, давления, крутящего момента и др. на основе магнитных цепей с распределенными параметрами.

Рассмотрены методы анализа потокораспределения в нелинейных магнитных цепях и даны примеры их практического использования для расчета различных типов датчиков. Показано, что на основе нескольких базовых моделей магнитных систем могут быть построены датчики механических величин различного назначения, обладающие конструктивным и схемотехническим единообразием, идентичными метрологическими показателями, нормализованными выходными сигналами в соответствии с требованиями ГСП.

Для инженерно-технических работников, занимающихся проектированием и эксплуатацией средств информационно-измерительной техники.

МикроЭВМ в информационно-измерительных системах / С.М. Переверткин, Н.И. Гаранин, Ю.М. Костин, И.И. Миронов. — 18 л.: ил. — (В пер.) 1 р. 20 к.

Рассмотрены вопросы анализа, синтеза и выбора оптимальных структур и параметров адаптивных информационно-измерительных систем (ИИС), построенных на основе использования микроЭВМ для обработки информации. Большое внимание уделено вопросам повышения точности, надежности, экономической эффективности и достоверности обработки измерительной информации. Для получения оптимальных решений в условиях недостатка априорной информации использованы методы непараметрической статистики и теории массового обслуживания.

Для инженерно-технических работников, занимающихся разработкой сложных многоканальных измерительных систем.

Приборы для хроматографии /К.И. Сакодынский, В.В. Бражников, А.Н. Буров и др. — 2-е изд., перераб. и доп. — 18 л.: ил. — (В пер.): 1 р. 30 к.

Изложены вопросы принципиального устройства газовых и жидкостных хроматографов. Указаны основные способы введения анализируемых смесей. Описаны термостаты и способы регулирования температурного режима, дозирующие устройства, универсальные и селективные детекторы, схемы соединения колонок и детекторов для сложных методик анализа, устройства обработки хроматографической информации.

В данном издании (1-е изд. 1973 г.) полностью переработаны и значительно обновлены сведения по жидкостным хроматографам, включен новый материал по аппаратуре для хроматографии в тонком слое, по дополнительным устройствам, расширяющим возможности анализа и др.

Для инженерно-технических работников, использующих газовую и жидкостную хроматографию для анализа и исследования сложных многокомпонентных смесей и химических соединений.

По всем вопросам приобретения новых книг, в том числе оформления предварительных заказов, читателям следует обращаться непосредственно в местные магазины, распространяющие научно-техническую литературу, а также в специализированные магазины — опорные пункты издательства "Машиностроение".

40 коп.



« МАШИНОСТРОЕНИЕ »