ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА И ПРИКЛАДНАЯ МАТЕМАТИКА

УДК 004.652.4

А.И. Баранчиков, Б.В. Костров

МЕТОДЫ ИССЛЕДОВАНИЯ МОДЕЛЕЙ И АЛГОРИТМОВ ПРЕДСТАВЛЕНИЯ ДАННЫХ ДЛЯ ПРЕДМЕТНЫХ ОБЛАСТЕЙ С РАНЖИРУЕМЫМИ АТРИБУТАМИ

Рассматривается подход к проектированию схем реляционных баз данных (РБД), основанный на получении дополнительной семантической информации о предметной области, получаемой на основе анализа (реинжиниринга) существовавших на момент исследования информационных систем (в части баз данных), являющихся отображением семантики предметной области, и проектировании схем новых баз данных с учетом полученной информации и дополнительных ограничений в виде ранжируемости атрибутов схемы базы данных, учитывающей распределенность данных и уровень их доступа.

Ключевые слова: *базы данных, реляционная, проектирование, верификация, предметная область, маскирование данных.*

Введение. Традиционный классический подход к проектированию схем РБД основан на следующих этапах (рассматривается упрощенная модель).

1. Анализ предметной области с определением семантических зависимостей между данными и объектами предметной области информационной системы.

2. Формализация полученных знаний.

3. Синтез схемы реляционной базы данных.

Однако на современном этапе информатизации общества крайне редко проектируемая база данных не имеет одного либо нескольких предшественников в виде баз данных, разработанных либо для всей предметной области новой информационной системы, либо для нескольких отдельных или пересекающихся частей. Другой аспект может быть связан с изменением самой предметной области, которая могла быть модифицирована в связи с изменением законодательства, правил бизнеса, наращиванием или сокращение функционала и др.

Любая грамотно построенная база данных должна адекватно отображать предметную область за счет корректной схемы базы данных, отражающей семантические зависимости данных в предметной области, и корректных данных, отражающих ее атрибутику.

Целью данной работы является получение методики проектирования схем РБД на основе выявления семантических зависимостей в процессе анализа предшествующих баз данных. Это позволит решить две важнейшие задачи.

1. Ускорение важнейшего этапа проектирования — анализа предметной области.

2. Верификация результатов анализа предметной области, который позволит повысить качество проектирования.

Предлагаемая методика проектирования схем реляционных баз данных представлена набором следующих методов.

1. Реинжиниринг схем РБД, который на основе схем, существовавших до начала проектирования РБД, позволяет получить формальное описание предметной области в виде множества семантических зависимостей (функциональных, многозначных и зависимостей соединения). При этом решается задача выявления атрибутов различного ранга.

2. Верификация полученного множества семантических зависимостей, которая позволит оценить соответствие схем РБД и хранящейся в

ней атрибутивной информации.

3. Сравнительный анализ моделей данных, полученных в результате экспертного анализа предметной области и реинжиниринга баз данных, в результате которого получается корректное описание предметной области на формальном языке реляционной алгебры.

4. Определение дополнительных ограничений, связанных с разделением доступа к информации и традиционными методами ее защиты на уровне отношений базы данных.

5. Проектирование схем РБД посредством алгоритмов, позволяющих учитывать дополнительные свойства атрибутов (конфиденциальность, ранжируемость и другие).

6. Модификация схем РБД для обеспечения дополнительной защиты данных на структурном уровне (например, маскирование реальных данных правдоподобными, но не правильными).

7. Верификация полученных схем РБД.

8. Построение окончательных схем РБД.

Рассмотрим подробнее основные этапы.

Реинжиниринг схем РБД. Под реинжинирингом в нашем случае следует понимать получение информации о семантике предметной области на основе анализа структуры базы данных, которая может быть получена путем анализа скриптов словаря данных и содержащейся в ней информации. При этом информация должна быть представлена в формальном виде, в котором ее можно впоследствии использовать для проектирования схемы новой РБД.

В простейшем виде база данных:

$$\boldsymbol{r}=(r_1(K_1,S_1),\ r_2(K_2,S_2),\ \dots\ r_i(K_i,S_i),\ \dots,\ r_k(K_k,S_k)$$

задается множеством отношений r_i, характеризуемым множеством ключей K_i и множеством атрибутов S_i, при этом можно сказать, что она представляется множеством функциональных зависимостей

$$F = \{\ K_1\to S_1,\ K_2\to S_2,\ \dots\ K_i\to S_i,\ \dots,\ K_k\to S_k\}.$$

Однако в процессе реинжиниринга реальных схем РБД следует учитывать несколько аспектов, которые требуют дополнительного внимания:

- адекватность схемы РБД семантике хранящихся в ней данных;
- количество БД, необходимых для проведения реинжиниринга;
- при исследовании нескольких баз данных следует анализировать существование в них разноименных атрибутов, соответствующих одним доменам;
- правильность определения ключей в отношениях баз данных;
- возможность «отставания» структуры базы данных от текущего состояния предметной области;
- возможность выявления атрибутов, обладающих дополнительными признаками (ранжируемых атрибутов), такими как конфиденциальность, признак территориальной принадлежности для распределенных систем и др.

Результатом реинжиниринга должна быть обобщенная модель предметной области, описанная формально в терминах реляционной алгебры, что позволит в дальнейшем использовать эту модель для решения задачи проектирования новой базы данных.

Пусть дано отношение r со схемой $R=\{A_1, A_2,\dots,A_n\},\ a_{ij}=t_j(A_i),\ j=1,2,\dots,m$. Тогда заданное отношение r будет удовлетворять некоторой функциональной зависимости $A_i\to A_k$, если проекция $\pi_{A_k}(\sigma_{A_i=a_{ij}}(r))$ содержит не более чем один кортеж для каждого A_i-значения a, т.е.

$$|\,\pi_{A_k}(\sigma_{A_i=a_{ij}}(r))|\le 1\,.$$

Алгоритмы, реализующие реинжиниринг, предложены в работах [1, 2].

Алгоритмы, целью которых является выявление ограничений, связанных с ранжируемостью атрибутов, частным случаем которой может являться территориальная распределенность данных, их конфиденциальность или секретность, предложены в работе [3].

Верификация полученного множества семантических зависимостей. На этапе верификации полученная в терминах реляционной алгебры модель анализируется на предмет наличия аномалий, которые искажают представление моделью предметной области и делают модель неадекватной. Здесь необходимо осуществить следующие проверки:

- правильность первичных ключей имеющихся отношений;
- выявить по возможности другие ключи, присущие отношениям базы данных;
- проверить приложимость ключевых зависимостей к отношениям баз данных, из которых они получены на этапе реинжиниринга;

При верификации используются два основные метода: проверка модели по множеству ограничений и проверка по данным.

С помощью прогонки табло проверяется на эквивалентность по множеству ограничений C. В терминах эквивалентности табло для успешной проверки выводимости зависимостей соединения необходимо выполнение условия $\boldsymbol{T}_R\equiv c\boldsymbol{T}_I$, где $\boldsymbol{T}_I$ – табло, состоящее из одной строки выделенных переменных. Буква c говорит о наличии множества всевозможных ограничений, применимых к табло. Эквивалентность $\boldsymbol{T}_1 \equiv c\boldsymbol{T}_2$ справедлива тогда и только тогда, когда функция *chase* $c(\boldsymbol{T}_1) \equiv$ *chase* $c(\boldsymbol{T}_2)$, т.е. когда финальное табло T_1 по алгоритму прогонки *chase* эквивалентно финальному табло T_2 [4]. Значит, достаточно выполнения условия *chase* $c(\boldsymbol{T}_R) \equiv$ *chase* $c(\boldsymbol{T}_I)$. Но так как *chase* $c(\boldsymbol{T}_I)=\boldsymbol{T}_I$, то *chase* $c(\boldsymbol{T}_R) \equiv \boldsymbol{T}_I$.

Следовательно, необходимым и достаточным условием проверки является наличие строки выделенных элементов в *chase* $c(\boldsymbol{T}_R)$.

Алгоритмы, реализующие верификацию модели, приведены в работе [4].

Сравнительный анализ моделей данных. В том случае, когда одновременно ведется как системный анализ предметной области, так и реинжиниринг старых баз данных, очередной задачей становится сравнение двух полученных моделей данных. Для их успешного сравнения требуется, чтобы они были формально описаны одним и тем же способом. Только тогда можно использовать алгоритмы сравнения этих моделей. Из известных способов формального описания наиболее подходящим является описание в терминах реляционной алгебры.

Перед алгоритмами сравнения стоят следующие задачи:

- выявление различий в формальном описании предметной области;
- представление различий в формальном виде, понятном для системного аналитика;
- анализ возможных причин возникновения различий;
- представление предложений по ликвидации этих различий.

В процессе сравнительного анализа следует определить причины возникновения различий, которые могут требовать различной реакции разработчика.

Не вдаваясь в подробности, можно выделить основные группы причин.

1. Старые базы данных неадекватно отображали состояние предметной области.

2. Старые базы данных не учитывают динамику изменения предметной области.

3. На этапе вновь проводимого системного анализа предметной области были допущены ошибки.

Алгоритм верификации не способен однозначно выявить, к какой группе причин относятся выявленные различия, однако он должен их четко и однозначно описать на своем выходе. Далее эксперт на основании своего опыта определят принадлежность ошибок к одной из групп и в соответствии с этим выполняет корректировку схемы РБД.

Относительно степени трудоемкости устранения ошибок их можно разделить:

- на легко устранимые, когда полученные при сравнении различия устраняются путем элементарных действий:
  - использование дополнительной декомпозиции отношений;
  - отмена лишней декомпозиции;
  - согласование имен атрибутов;
- трудно устранимые, когда полученные различия требуют дополнительного анализа и формальными методами практически не устраняются.

В последнем случае следует прийти к решению, связанному с определением «виновника» полученных расхождений, и выбрать правильное решение.

В результате выполнения алгоритмов этого этапа будет получено уточненное формальное описание предметной области, которое является основой для проектирования схемы РБД.

Определение дополнительных ограничений, связанных с разделением доступа к информации и традиционными методами ее защиты на уровне отношений базы данных. Современные информационные системы очень чувствительны к проблеме информационной безопасности и распределенности данных. Несмотря на это, существующие методики проектирования схем РБД не учитывают этот аспект и традиционно трактуют атрибуты БД как некоторую сущность, не обладающую определенными свойствами, которые в процессе проектирования появляются только на этапе физического проектирования базы данных, когда атрибуту присваиваются тип, размерность и так далее.

Однако уже на этапе логического проектирования возникает необходимость добавления признаков атрибутов, по которым можно будет детализировать их принадлежность к определенным отношениям, которые обособляются в отдельные отношения в силу требований к распределению доступа либо территориальной распределенности. Такие признаки можно определить как ранжируемость атрибутов. Наличие признаков ранжируемости позволяет сформулировать дополнительные требования к проектируемой схеме РБД. Например: в одном отношении могут содержаться ранжируемые атрибуты только одного ранга или в конкретном отношении могут содержаться атрибуты не выше (ниже) определенного ранга. Алгоритмы приведены в работах [5,6]

Проектирование схем РБД посредством алгоритмов, позволяющих учитывать дополнительные свойства атрибутов (конфиденци-

альность, ранжируемость и другие). Теперь перед проектировщиком базы данных встает задача — каким образом можно учесть дополнительные свойства атрибутов, которые позволят получить корректную базу данных, адаптированную к этим свойствам. Этого можно добиться путем модификации классических алгоритмов проектирования схем РБД. Наиболее распространенными из них являются алгоритм синтеза и алгоритм декомпозиции. На входе как одного, так и другого алгоритма имеется множество семантических зависимостей, представленных в терминах реляционной алгебры, а на выходе — схемы РБД в виде множества нормализованных отношений и множества ключей для каждого отношения. Дополнительные ограничения вводятся в виде рангов атрибутов, под которыми мы в данном случае понимаем свойства атрибутов, которые требуют отнесения атрибутов различного ранга к различным отношениям базы данных в одном случае либо запрет размещения атрибутов более высокого ранга в отношениях, содержащих ранжируемые атрибуты низкого ранга.

Для каждого входного ключевого атрибута K_i схемы R вводится избыточность в виде атрибутов K_i', таких что $K_i \to K_i'$ и $K_i' \to K_i$, то есть $K_i \leftrightarrow K_i'$; $K_i \in K$, $K_i' \in K'$, где K – множество входных ключевых атрибутов, K' - множество атрибутов, введённых для однозначного определения ключевых входных атрибутов.

На выходе модернизированных алгоритмов будут получены схемы РБД, удовлетворяющие ограничениям, накладываемым введением дополнительных свойств атрибутов.

Алгоритмы приведены в работе [7].

Модификация схем РБД для обеспечения дополнительной защиты данных на структурном уровне (например, маскирование реальных данных правдоподобными, но не правильными). Для более сильной защиты информации может потребоваться дополнительная модификация схемы базы данных, которая позволит реализовать такой механизм, как маскирование данных, позволяющий за счет внедрения маскирующей информации (ложной информации, которая заменяет для потенциального нарушителя информацию действительную, которую он стремится получить), позволяющей снизить активность нарушителя за счет введения его в заблуждение в том, что он уже добился своей цели.

Если разбить субъекты маскировки на группы, которым доступ предоставлен одинаково, то, учитывая человеческий фактор, можно предположить обмен данными между такими пользователями. Пусть существуют пользователи *a* и *b*, имеющие одинаковые настройки доступа к данным:

$$r_a = \sigma_{mask(a)}(r);$$
$$r_b = \sigma_{mask(b)}(r).$$

Если данные r_a и r_b будут отличаться, то в производственном процессе этот факт может быть обнаружен, в результате чего пользователи смогут понять, что существует система маскировки данных, важной задачей которой является необнаружимость для пользователей.

Схема РБД модифицируется таким образом, что, во-первых, облегчается формирование различных пользовательских представлений для реализации механизма маскирования данных и, во-вторых, достигается возможность распределения участвующих в маскировании данных по различным отношениям. Последнее позволяет добиться более высокой защищенности маскируемых данных за счет отнесения содержащих их отношений к другим пользовательским схемам (базам данных), доступ к которым невозможен для большинства легальных пользователей информационной системы.

При построении систем, использующих маскирование данных, требуется решить еще одну важную задачу, суть которой заключается в том, что пользователи, для которых предназначено маскирование, в свою очередь, в меру своей компетенции должны воспринимать маскируемую информацию как правдоподобную, иначе они будут предпринимать дополнительные попытки добычи реальной информации, что значительно снизит эффективность маскирования данных.

Другим вариантом реализации маскирования данных является отвлечение злоумышленника за счет предоставления ему заведомо неправдоподобной информации в той области, где отсутствуют данные, требующие дополнительной защиты.

Маскирование данных хорошо тем, что для повышения уровня безопасности хранимой информации не требуется дополнительных материальных и организационных затрат.

Верификация полученных схем РБД. Полученная на предыдущих этапах схема базы данных обязательно должна подвергнуться верификации, которая предусматривает два подхода: верификация полученной схемы РБД по множеству ограничений, представленному в виде набора семантических зависимостей (F-зависимостей, MV-зависимостей и J-зависимостей), и верификация по репрезентативному набору данных. Положительное выполнение обеих этих проверок позволяет сделать вывод об адекватном отображении предметной области схемой РБД.

Построение окончательных схем РБД. На последнем этапе остается выполнить задачу преобразования полученной и проверенной схемы РБД, представленной в терминах реляционной алгебры, в скрипт на языке SQL для формирования физической структуры базы данных. Здесь следует учитывать то, что имеются некоторые различия при описании структур данных в различных реляционных СУБД.

Экспериментальные исследования. Алгоритмы, приведенные в работах [1-7], были исследованы теоретически. Результаты теоретических исследований, которые включали проверку сходимости и оценку временной сложности алгоритмов, показали, что все исследованные алгоритмы сходятся, а их временная сложность позволяет решать поставленные задачи в разумное время.

Экспериментальное исследование приведенных алгоритмов связано с определенными сложностями по следующим причинам:

- предприятия и организации крайне неохотно предоставляют (не предоставляют вообще) реальные базы данных для проведения экспериментальных исследований алгоритмов;
- банк баз данных для проведения исследований недостаточен и не может предоставить репрезентативную выборку для проведения качественных исследований;
- возникают трудности исследования алгоритмов применительно к базам данных, которые удовлетворяют определенным набором критериев.

Все вышесказанное привело к необходимости разработки специального инструментария, который позволяет генерировать тестовые базы для проведения экспериментального исследования разработанных алгоритмов [8].

Тестовые базы данных генерируются с учетом ряда специально выявленных критериев, которые определяют класс предметной области, зависящий от сложности семантической организации предметной области, количества присущих ей атрибутов и мощности доменов этих атрибутов.

Сгенерированный набор данных должен включать множество атрибутов A = {A1, A2, …, Ag}; множество семантических зависимостей U, включающих данные атрибуты, состоящее из подмножеств F-зависимостей F, MV-зависимостей MV и J-зависимостей J, причём $U=F\cup MV\cup J$; множество признаков ранжируемости Y = {Y1, Y2, … , Yn} и соответствующие множества атрибутов каждого признака $\mathbf{A}_{Y_1}$, $\mathbf{A}_{Y_2}$, ..., $\mathbf{A}_{Y_n}$, причем

$$\mathbf{A} = \mathbf{A}_{Y_1} \cup \mathbf{A}_{Y_2} \cup \ldots \cup \mathbf{A}_{Y_n} \,.$$

Генерация тестовых баз данных позволила получить репрезентативную выборку, достаточную для проведения качественных экспериментов, подтвердивших теоретические исследования разработанных алгоритмов.

Выводы. Полученные результаты носят прикладной характер.

Предложена методика проектирования схем реляционных баз данных, отличающаяся от традиционных возможностью использования знаний о предметной области проектируемой базы данных, определяемых на основе анализа структур и данных предшествующих баз данных, разработанных для этой же предметной области, и использования дополнительных ограничений, связанных с ранжируемостью атрибутов.

Разработанная методика позволяет, **во-первых**, сократить критический путь проекта разработки информационных систем за счет сокращения времени анализа предметной области, **во-вторых**, снизить количество наиболее критичных ошибок, возникающих в процессе анализа предметной области за счет проведения сравнительного анализа реинжиниринга предшествующих баз данных и результатов «ручного» системного анализа, и, **в-третьих**, повысить безопасность данных на структурном уровне базы данных за счет учета дополнительных ограничений на атрибуты базы данных и маскирования данных.

Библиографический список

1. *Баранчиков А.И., Алпатова А.В.* Реинжиниринг ключей в отношениях реляционных баз данных // Вестник РГРТУ. № 4 (выпуск 26). Рязань, 2008.

2. *Баранчиков А.И., Асташина Е.А.* Модифицированный алгоритм реинжиниринга функциональных зависимостей // Вестник РГРТУ № 4 (выпуск 34). Рязань: РГРТУ. 2010. С.55-58.

3. *Баранчиков А.И., Кухарев С.Э.* Алгоритмы реинжиниринга атрибутов конфиденциальности в реляционных базах данных // Вестник РГРТУ. № 1 (выпуск 35). Рязань. РГРТУ. 2011. С.72-75.

4. *Баранчиков А.И., Дрожжин И.В.* Проверка структуры базы данных на правильность логического построения с использованием табло // Информационные системы и технологии. № 3 (71). Орел: Госуниверситет – УНПК. 2012. – С. 9-14.

5. *Баранчиков А.И., Баранчиков П.А., Пылькин А.Н.* Алгоритмы и модели ограничения доступа к записям баз данных. - М.: Горячая линия — Телеком, 2011. - 182 с.

6. *Баранчиков А.И., Баранчиков П.А.* Методы маскировки данных в отношениях БД для различных

моделей доступа // Системы управления и информационные технологии: науч.-техн. журн. № 2(52). Москва - Воронеж, 2013. С. 58 – 61.

7. *Баранчиков А.И., Громов А.Ю.* Построение схем реляционных баз данных с n-ранжируемыми атрибутами // Системы управления и информационные технологии: науч.-техн. журн. № 2.1(48). Москва – Воронеж. 2012. С.114-117.

8. *Баранчиков А.И., Громов А.Ю.* Алгоритм генерации формализованной модели предметной области // Вестник РГРТУ: науч.-техн. журн. № 3 (выпуск 33). Рязань. 2010. С. 45-49.

УДК 629.1

К.В. Миронова

ОЦЕНКА НАДЕЖНОСТИ ОБЩЕГО МЕТОДА НАИМЕНЬШИХ КВАДРАТОВ В ЗАДАЧАХ ПРОЕКТИРОВАНИЯ УПРАВЛЕНИЯ

Предложена модернизация общего метода наименьших квадратов (ОМНК) с целью увеличения точности оценивания параметров механических объектов, описываемых дифференциальными уравнениями с управлением. Получены закон распределения отклонения оценок от истинных значений параметров и закон распределения отклонения оценочной дисперсии отклонения от номинального её значения на основе распределений Стьюдента и закона «хи квадрат».

Ключевые слова: *общий метод наименьших квадратов, законы распределения оценок, проектирование управления.*

Введение. Классическая задача проектирования оптимального управления механическими, в частности космическими, объектами выглядит следующим образом (см., например, [1]): состояние движущегося объекта определяется вектором фазовых координат $x=(x_1,\ldots,x_n)$, которые изменяются во времени, подчиняясь при этом дифференциальному уравнению векторного типа:

$$\dot{x}(t)=f(t,x,u,\varepsilon)\,, \qquad (1)$$

где фазовые координаты суть функции времени t , $f(\ldots)$ – векторная функция, координаты которой определяются конструкцией объекта и законами его движения, u – вектор управляющих воздействий (команд) на объект, который выбирается из заданного множества

$$u\in U_a\;, \qquad (2)$$

которое, в свою очередь, зависит от набора параметров $a=(a_1,\ldots,a_k)$, вектор ε обозначает внешние возмущения, действующие на объект. Начальное состояние объекта задается условием:

$$x(t_0)=x_0\,, \qquad (3)$$

где x_0 – априори заданный вектор.

Цель проектирования управления состоит в том, чтобы во время движения объекта на основе косвенных данных оценить вектор a , сформировать множество управлений U_a и выбрать из него такие управления $u\in U_a$, чтобы перевести объект, подчиненный условиям (1), (2), из точки фазового пространства (3) за конечное время $t\in[t_0,T]$ в заданную точку:

$$x(t)=x_T \qquad (4)$$

оптимальным (наилучшим) способом.

Оптимальность понимается традиционно для подобных задач: необходимо, чтобы функционал качества управления:

$$\Phi=\int_{t_0}^{T} g(t,x,u)\,dt \;\to\min\;,$$

здесь $g(t,x,u)$ – заданная функция описанных выше аргументов.

В рассматриваемых авторами задачах проектирования координаты вектора a оцениваются на основе косвенных наблюдений линейного вида, а точность оценивания играет зачастую ключевую роль в определении множества управлений $u\in U_a$.

Цели работы. 1. Модернизировать общий метод наименьших квадратов (ОМНК) с целью увеличения точности оценивания параметров.

2. Описать законы распределения оценок параметров для контроля качества увеличения точности оценивания.

Постановка задач. Предполагается, что имеется некоторая линейная, однородная по функциональным переменным $x_1(t),\ldots,x_n(t)$ (совпадающими с фазовыми переменными только по названию) зависимость их с измеряемой переменной

$$y(t) = \sum\nolimits_{j=1}^{n} a_j x_j(t)\,, \qquad (5)$$

коэффициенты которой a_j неизвестны и подлежат оцениванию (для последующего определения множества управлений U_a).

Для определения координат a_j в некоторые моменты времени $t_1,\ldots,t_N$, такие, что $t_1 < t_2 < \ldots < t_N < t_0$, производятся измерения величин:

$$y_r = \sum\nolimits_{j=1}^{n} a_j x_{jr}\,, \qquad (6)$$

где $y_r = y(t_r)$, $x_{jr} = x_j(t_r)$. Предполагается с учетом «физики» решаемых задач, что коэффициенты a_j однозначно определяются значениями x_{jr} и y_r , и, следовательно, ранг матрицы

$$rank\ X = rank\,[x_{jr}] \ge n\,. \qquad (7)$$

Из (7) мгновенно следует, что

$$N \ge n\,. \qquad (8)$$

Экспериментальные значения y_r сопряжены с неизбежными ошибками, и вместо истинных значений y_r получаются приближенные значения:

$$\eta_r = y_r + \Delta_r\,,\ r = 1,\ldots,N\,. \qquad (9)$$

Требуется: 1) по данным x_{jr} из (6) и данным η_r из (9) определить (оценить) рациональным образом по ОМНК приближенные значения $\hat{a}_j$ коэффициентов a_j ;

2) исходя из постулата о нормальном законе распределения ошибок Δ_r с известной матрицей корреляций, найти закон распределения отклонений $\hat{a}_j - a_j$ в случае ОМНК.

Ретроспектива общего метода наименьших квадратов. Пусть для некоторых оценок $\hat{a}_j$ значения

$$\hat{\eta}_r = \sum\nolimits_{j=1}^{n} \hat{a}_j x_{jr}\,,\ \varepsilon_r = \eta_r - \hat{\eta}_r\,. \qquad (10)$$

Следуя идеологии ОМНК (основы которого заложены в работах Гаусса и Лежандра, метод усовершенствован в работах Эдрейна, Энна, Бесселя, Ганзена, из отечественных крупных результатов укажем на работы [2, 3] и обобщающую монографию [4]), оценки $\hat{a}_j$ определяют из условия:

$$(\varepsilon^T W, \varepsilon) \to \min\,, \qquad (11)$$

где вектор-столбец ε формируется естественным образом, W – это заданная весовая, симметрическая, положительно определенная матрица, $\varepsilon^T W$ есть вектор-строка (здесь T – символ транспонирования), а $(\ldots,\ldots)$ – обозначение скалярного произведения векторов.

Известно (см., например, [5]), что матрицу W можно разложить в произведение матриц $W = P^T P$, где P – невырожденная квадратная матрица. Эффективный алгоритм поиска матрицы P на основе блочного метода Гаусса предложен в работе [6].

Тогда обобщенная сумма квадратов невязок может быть представлена (с переходом на матричную запись) в виде

$$\varepsilon^T W \varepsilon = \varepsilon^T P^T P \varepsilon = (P\varepsilon)^T (P\varepsilon)\,.$$

Последнее означает, что в задаче поиска экстремума (11) минимизируется величина

$$[P(\eta - X\hat{a})]^T [P(\eta - X\hat{a}] = [P\eta - PX\hat{a}]^T [P\eta - PX\hat{a}] =$$

$$= (\eta^* - X^*\hat{a})^T (\eta^* - X^*\hat{a}) = (\varepsilon^*, \varepsilon^*)\,, \qquad (12)$$

при естественных новых обозначениях.

Таким образом, ОМНК удается свести с помощью некоторого линейного преобразования к классическому МНК. Из (9) следует, что

$$\eta^* = Py - P\Delta\,, \qquad (13)$$

где вектор ошибок Δ записан как столбец.

Известно [5], что преобразованная таким образом модель измерений удовлетворяет условиям теоремы Гаусса – Маркова.

Как следствие, МНК-оценки будут наиболее эффективными в классе всех линейных несмещенных оценок, если в качестве матрицы W взять матрицу, обратную ковариационной матрице V ошибок (теорема Айткена):

$$\hat{a} = (X^T V^{-1} X)^T (X^T V^{-1})\eta\,. \qquad (14)$$

Законы распределения для числовых характеристик ОМНК. На практике для оценки надежности получаемых по МНК оценок (вычисление вероятности ошибок, превосходящих заданные пределы), как правило, исходят из закона Гаусса, описывающего распределение отклонений $\hat{a}_j - a_j$. Это приводит, как доказано в работе [3], к существенной переоценке этой надежности. Одновременно можно дать более надежные оценки отклонений $\hat{a}_j - a_j$ для ОМНК, учитывая только что проведенное сведение ОМНК к МНК.

Действительно, пусть случайные ошибки измерений Δ подчинены закону Гаусса со средним, равным нулю, и единичной дисперсией (нормированное распределение).

Пусть матрица $P = [p_{ij}]$ (индексы i и j пробегают естественные значения).

Из равенства (13) следует, что ошибки

$$\Delta^* = P\Delta = (\Delta_1^*, \ldots, \Delta_N^*)^T$$

преобразованной с помощью матрицы P модели измерений (9) также подчинены закону Гаусса со средним, равным нулю, и дисперсией

$$D(\Delta_j^*) = \sum\nolimits_{k=1}^{N} p_{jk}^2 . \qquad (15)$$

Нормируем модель измерений (13), разделив каждую j-ю строку на дисперсию (15). Получим классическую для МНК нормированную модель измерений:

$$\overline{\eta} = \overline{y} + \overline{\Delta} \qquad (16)$$

при $\overline{y}_r = \sum\nolimits_{j=1}^{n} a_j \overline{x}_{jr}$, $\overline{\eta}_r = \overline{y}_r + \overline{\Delta}_r$, $r = 1, \ldots, N$ и естественных обозначениях векторов, ошибки которой подчинены закону Гаусса.

Найдем классические МНК-оценки $\hat{a}_j$ компонент a_j по модели (16).

Известно (см. [2] и [4]), что полученные приближенные значения $\hat{a}_j$ не содержат «систематической ошибки»: математическое ожидание:

$$M[\hat{a}_j] = a_j .$$

Первичная оценка точности этих приближений задается формулой для средней квадратической ошибки $\sigma[\hat{a}_j]$:

$$\sigma^2[\hat{a}_j] = M[\hat{a}_j - a_j]^2 = q_{jj} s^2 ,$$

где s – это средняя квадратическая ошибка экспериментального определения величины $\overline{y}_r$ (она уже не зависит от номера r), а q_{ir} определяются из уравнения:

$$\sum\nolimits_{j=1}^{n} [\overline{x}_i \overline{x}_j]\, q_{ir} = e_{ir} ,\ i, r = 1, \ldots, n ,$$

где e_{ir} – символ Кронекера.

Если величина s *a priory* неизвестна, то обычно (см. [2] и [4]) ее принимают приближенно равной:

$$\sigma = \sqrt{\frac{(\overline{\varepsilon}, \overline{\varepsilon})}{N - n}} ,$$

где координаты вектора $\overline{\varepsilon}$ формируются по аналогии с (10) для вектора ε, но уже для нормированной модели измерений (16).

Известно (см. [2] или [4]), что математическое ожидание

$$M[\sigma^2] = s^2 .$$

Главный вопрос всякой задачи оценивания состоит в том, чтобы указать закон распределения отклонения оценки $\hat{a}_j$ от истинного значения параметра a_j.

На этот вопрос для ОМНК можно получить ответ, сведя описанным образом задачу ОМНК к нормированной модели (16), и применить основные результаты работы [2] для МНК.

Учитывая проведенные выше рассуждения и построения, связанные со сведением ОМНК к классическому МНК, можно считать доказанными следующие две теоремы.

Теорема 1. При сведении описанным образом ОМНК к МНК оценка отклонения величины σ от величины s для модели (16) задается формулой:

$$P_v\left((N - n)\frac{\sigma^2}{s^2} < h^2 \right) = H_{N-n}(h^2) ,$$

где $P_v(\ldots)$ – это вероятность соответствующего события, а для распределения «хи квадрат»:

$$H_m(h^2) = \frac{1}{\frac{m-2}{2}\Gamma\left(\frac{m}{2}\right)} \int\limits_0^h h^{m-1} e^{-\frac{h^2}{2}} dh ,$$

а $\Gamma(\ldots)$ – это гамма-функция Эйлера.

Теорема 2. При сведении описанным образом ОМНК к МНК оценка отклонения оценки $\hat{a}_j$, полученной на основе модели (16), от истинного значения a_j задается формулой:

$$P_v\left(\frac{\hat{a}_j - a_j}{\sqrt{q_{jj}}\,\sigma} < t \right) = S_{N-n}(t) ,$$

а для распределения Стьюдента:

$$S_m(t) = \frac{\Gamma\left(\frac{m+1}{2}\right)}{\sqrt{m\pi}\,\Gamma\left(\frac{m}{2}\right)} \int\limits_{-\infty}^{t} \left(1 + \frac{t^2}{m}\right)^{-\frac{m+1}{2}} dt .$$

Вывод. Модернизирован общий метод наименьших квадратов (ОМНК) с целью увеличения точности оценивания параметров. В работе определены рациональным образом по ОМНК приближенные значения $\hat{a}_j$ коэффициентов a_j. Найден закон распределения отклонений $\hat{a}_j - a_j$ в случае ОМНК. Описаны законы распределения оценок параметров для контроля качества увеличения точности оценивания.

Библиографический список

1. *Атанс М., Фалб П.* Оптимальное управление /Пер. с англ. Ю.И. Тютчева. – М.: Машиностроение, 1968. – 764 с.

2. *Колмогоров А.Н.* К обоснованию метода наименьших квадратов //Успехи математических наук. 1946. Т. 1. Вып. 11. С. 57–70.

3. *Колмогоров А.Н., Петров А.А., Смирнов Ю.М.* Одна формула Гаусса из теории метода наименьших квадратов //Известия АН СССР. Серия математическая. 1947. Т. 11. С. 561–566.

4. *Линник Ю.В.* Метод наименьших квадратов и основы статистической теории обработки наблюдений. – М.: Наука, 1962. – 334 с.

5. *Гантмахер Ф.Р.* Теория матриц. – М.: Наука, 1966. – 576 с.

6. *Миронова К.В.* Блочный метод Гаусса в задачах роектирования линейного управления. 6-я межд. науч.-практ. конф. «Космонавтика. Радиоэлектроника. Геоинформатика»: тез. докл. / РГРТУ. Рязань. 2013. С. 107–108.

УДК 517.977.5

А.В. Кузнецов

ЧИСЛЕННОЕ РЕШЕНИЕ ЗАДАЧИ ОПТИМАЛЬНОГО УПРАВЛЕНИЯ ГРАДИЕНТНЫМ МЕТОДОМ

Для задач оптимального управления процессами, моделируемыми системами обыкновенных дифференциальных уравнений с кусочно-постоянным управлением, применен градиентный поиск искомого управления с проекцией в допустимое множество.

Ключевые слова: *задача оптимального управления, градиентный поиск, кусочно-постоянное управление.*

Введение. Управление техническими системами детерминированного типа в некоторых случаях не может быть произвольного вида. На это оказывает влияние инерционность механизма, его прочностные характеристики и т.п. Тем не менее, задача экономичного в некотором смысле управления подобными системами требует решения. Одним из классов подобного ограниченного управления являются кусочно-постоянные управления. Поэтому возникает необходимость исследования задач оптимального управления на данном, более узком, в сравнении с кусочно-непрерывным классом управлений, множестве. В работе применен численный метод градиентного типа для поиска локально оптимального управления. В рассматриваемых задачах управление принадлежит классу кусочно-постоянных векторных функций; оптимизируемыми параметрами являются сами векторные значения управлений и моменты переключения управлений. При этом указанные моменты не предполагаются известными и подлежат оптимизации. Подобная постановка задачи оптимального управления на классе кусочно-постоянных функций исследовалась ранее [1–6]. Так, в работе [2] использована фундаментальная матрица решений для поиска кусочно-постоянных управляющих функций в решении линейно-квадратичной задачи управления. В [4] исследовался случай, когда управления принимают значения из заданного конечного множества значений. В работе [6] для построения последовательности приближений использован метод сопряженной задачи.

Новизна результатов данного исследования состоит в том, что количество интервалов постоянства управляющей функции заранее фиксировано, исходя из практических соображений или технических требований, и может быть только уменьшено, что представляется положительным фактором.

Целью статьи является получение явных зависимостей для градиента критерия качества (продолжение исследований автора [7]), что позволяет использовать их для оптимизации управления на классе исключительно кусочно-постоянных функций**.**

Рассмотрим задачу оптимального управления, моделируемую системой обыкновенных дифференциальных уравнений с управлением. Пусть состояние управляемой системы задано в нормальной форме (векторный вид):

$$\dot{x}(t) = f(t,x,u)\,,\ t \in (0,T]\,,\ x(0) = x_0\,, \qquad (1)$$

где $x = x(t)$, $t \in [0,T]$ – n-мерный вектор состояния объекта; $u = u(t)$, $t \in [0,T]$ – m-мерное управление объектом; вектор-функция f непрерывна по совокупности переменных (x,u) вместе со своими первыми частными производными. Длительность процесса T и начальное состояние системы x_0 фиксированы.

Управление $u = u(t)$ постоянно на каждом полуинтервале $[\tau_{j-1}, \tau_j)$, $j = 1,..,K$, полученном разбиением отрезка $[0, T]$ на K полуинтервалов с помощью моментов переключения τ_j, $j = 1,..,K-1$,

$$u(t) = v_j \ , \ t \in [\tau_{j-1}, \tau_j) \ , \ \tau_{j-1} \le \tau_j \ , \ \ j = 1,..,K \ ,$$
$$\tau_0 = 0 \ , \ \tau_K = T \ , \qquad (2)$$

а m-мерные значения управления v_j, $j = 1,..,K$ принадлежат некоторому замкнутому ограниченному множеству U, к примеру, параллелепипеду:

$$U = \{v = (v_1, v_2, .. v_K) : B_j^i \le v_j^i \le A_j^i, i = 1,..m, j = 1,..K\}. \qquad (3)$$

Задача оптимизации заключается в нахождении значений управления $u = u(t)$ и границ интервалов постоянства этих значений, определяемых вектором $\tau = (\tau_1, \tau_2, ..., \tau_{K-1})$, при которых заданный функционал

$$\widetilde{J}[u, x] = J(v, \tau) = \int_0^T f^0(t, x, u) dt \ , \qquad (4)$$

где $(v, \tau) \in R^{Km+K-1}$, при условиях (1) - (3) принимает минимальное значение. Подынтегральная функция f^0 непрерывна вместе со всеми своими частными производными по всем аргументам и число интервалов постоянства управления K фиксировано.

Схема численного решения задачи. Для численного решения задачи, следуя схеме, описанной в [6], на отрезке $[0, T]$ вводится равномерное разбиение

$$\Omega = \{t_i : t_i = ih, i = 0,..,N, h = T / N\},$$

где N - заданное натуральное число, h - шаг дискретизации. Систему (1) аппроксимируем явной схемой Эйлера:

$$x_{i+1} = x_i + hf(t_i, x_i, u_i) \ , \ i = 0,...,N-1, \qquad (5)$$

где x_0 определено из задачи (1).

Аппроксимируем значение управления $u = u(t)$, $t \in [0, T]$ на множестве Ω следующим образом:

$$u_i = \begin{cases} v_j, [t_i, t_{i+1}) \subset [\tau_{j-1}, \tau_j), \\ \frac{1}{h}\left(v_{j+1}(t_{i+1} - \tau_j) + v_j(\tau_j - t_i)\right), \tau_j \in [t_i, t_{i+1}), \end{cases} \qquad (6)$$

где $j = 1..K-1, i = 0..N-1$.

Для вычисления функционала качества (4) воспользуемся квадратурной формулой:

$$J(v, \tau) = h \sum_{i=0}^{N-1} f^0(t_i, x_i, u_i) \to \min \ . \qquad (7)$$

Получена конечная задача математического программирования (7) с учетом условий (2), (3), (5) и (6).

Для решения задачи (7) при условиях (5),(6), для определения оптимальных значений векторов v и τ, используем градиентный метод конечномерной оптимизации функционала $J(v, \tau)$, учитывая ограничения на (v, τ):

$$\begin{pmatrix} v^{k+1} \\ \tau^{k+1} \end{pmatrix} = \mathrm{P}\left[\begin{pmatrix} v^k \\ \tau^k \end{pmatrix} - \delta \begin{pmatrix} \partial J(v^k, \tau^k)/\partial v \\ \partial J(v^k, \tau^k)/\partial \tau \end{pmatrix}\right], \qquad (8)$$

где $k = 0,1,2,...$, P – проектор вектора параметров (v, τ) на множество параметров с учетом (2),(3); (v^0, τ^0) – начальное приближение для параметров из допустимого множества;

$$\partial J/\partial v = (\partial J/\partial v_1, ..., \partial J/\partial v_K),$$
$$\partial J/\partial \tau = (\partial J/\partial \tau_1, ..., \partial J/\partial \tau_{K-1}), \qquad (9)$$

являются компонентами градиента функционала задачи (7).

Рассмотрим вспомогательные векторы сопряженных переменных:

$$p_i = dJ(v, \tau)/dx_i, \ \ p_i \in R^n \ , \ i = 0,...,N \ , \qquad (10)$$

где производная вычисляется с учетом зависимости между значениями x_i, $i = 0,..,N$, определенными в (5). Из вида функционала (7) с учетом равенств (5) имеем:

$$p_i = \frac{\partial J}{\partial x_i} + \frac{\partial J}{\partial x_{i+1}} \frac{\partial x_{i+1}}{\partial x_i} = h \frac{\partial f^0(t_i, x_i, u_i)}{\partial x_i} + p_{i+1}\left[E_n + h \frac{\partial f(t_i, x_i, u_i)}{\partial x_i}\right]$$
$$p_N = 0 \ , \ i = N-1,..,0 \ , \qquad (11)$$

где E_n – n-мерная единичная матрица. Система относительно переменных p_i называется сопряженной.

В общем случае момент τ_j находится между узловыми точками $t_{k_j - 1}$ и t_{k_j}, $\tau_j \in [t_{k_j - 1}, t_{k_j})$, $j = 1,..,K-1$. Тогда величины $\partial J/\partial v_j$, $j = 1,..,K$ определяются следующим образом:

$$\frac{dJ}{dv_j} = \frac{\partial J}{\partial v_j} + \sum_{s=k_{j-1}}^{k_j} p_s \frac{\partial x_s}{\partial v_j} =$$
$$= h \sum_{s=k_{j-1}}^{k_j} \left(\frac{\partial f^0(t_{s-1}, x_{s-1}, u_{s-1})}{\partial u_{s-1}} \frac{\partial u_{s-1}}{\partial v_j} + p_s \frac{\partial f(t_{s-1}, x_{s-1}, u_{s-1})}{\partial u_{s-1}} \frac{\partial u_{s-1}}{\partial v_j} \right),$$
$$j = 1,...,K \ , \qquad (12)$$

а величины $\partial u_s/\partial v_j$, $s = k_{j-1} - 1,...,k_j - 1$,

$j = 1,...,K$ находятся из соотношения (6):

$$\frac{\partial u_s}{\partial \nu_j} = \begin{cases} 1, s = k_{j-1},...,k_j - 2, \\ \left(t_{s+1} - \tau_{j-1}\right)/h, s = k_j - 1, \\ \left(\tau_j - t_s\right)/h, s = k_j. \end{cases}$$

Элементы $\partial J/\partial \tau_j$ из (9) $j = 1,..,K-1$ представляются в виде

$$\frac{\partial J}{\partial \tau_j} = \frac{\partial J}{\partial \tau_j} + p_{k_j}\frac{\partial x_{k_j}}{\partial \tau_j} = \frac{\partial J}{\partial \tau_j} + p_{k_j}\frac{\partial x_{k_j}}{\partial u_{k_j-1}}\frac{\partial u_{k_j-1}}{\partial \tau_j} =$$

$$= h\frac{\partial u_{k_j-1}}{\partial \tau_j}\left[\frac{\partial f^0(t_{k_j-1}, x_{k_j-1}, u_{k_j-1})}{\partial u_{k_j-1}} + p_{k_j}\frac{\partial f(t_{k_j-1}, x_{k_j-1}, u_{k_j-1})}{\partial u_{k_j-1}}\right], \quad (13)$$

$j = 1,..,K-1$.

Производные $\partial u_{k_j-1}/\partial \tau_j$, $j = 1,..,K-1$ имеют вид, с учетом (6):

$$\frac{\partial u_{k_j-1}}{\partial \tau_j} = \left(\nu_j - \nu_{j+1}\right)/h, \; j = 1,..,K-1. \quad (14)$$

Тогда из (13),(14) для $j = 1,..,K-1$ имеем:

$$\frac{\partial J}{\partial \tau_j} = \left(\nu_j - \nu_{j+1}\right)\left[\frac{\partial f^0(t_{k_j-1}, x_{k_j-1}, u_{k_j-1})}{\partial u_{k_j-1}} + p_{k_j}\frac{\partial f(t_{k_j-1}, x_{k_j-1}, u_{k_j-1})}{\partial u_{k_j-1}}\right]. \quad (15)$$

С учетом вышеуказанного градиент функционала задачи (7) полностью определен.

Поиск локального экстремума для задачи (7) состоит в последовательном выполнении шагов метода (8):

1) по параметрам $\left(\nu^k, \tau^k\right)$ по формулам (5), (6) строится решение дискретной задачи $\{x_i\}_{i=0}^N$;

2) решается сопряженная система (11) в обратном порядке, начиная с p_{N-1} до p_0;

3) из соотношений (12), (15) определяем градиент функционала (9);

4) выполняется шаг метода (8) с выбором δ из условия одномерной минимизации функционала (7) и определяется новое приближение $\left(\nu^{k+1}, \tau^{k+1}\right)$.

Замечание. Как было замечено [5], в случае плохо обусловленных задач (так называемая «овражность») выбор точного минимума на шаге 4 может приводить из-за погрешности вычислений к неэффективным шагам вблизи стационарных точек. Для исправления указанного эффекта предлагается в подобных случаях заменить шаг 4 на дихотомический процесс с масштабным параметром, варьируя который можно управлять скоростью сходимости.

Этапы 1 – 4 повторяются до тех пор, пока не будут выполнены условия остановки. Остановка произойдет либо при достижении порога малости модуля вектора градиента, либо при исчерпании максимального числа итераций.

Вычислительный эксперимент. Рассмотрим модель колебательного контура с изменяемыми характеристиками (аналог уравнения Хилла). На промежутке $[0, 3\pi/4]$ [6] ставится задача максимизировать амплитуду в конечной точке отрезка. Соответствующая математическая постановка задачи имеет вид:

$$\begin{cases} \dot{x}_1 = x_2, \\ \dot{x}_2 = -ux_1, \end{cases} \quad \begin{cases} x_1(0) = 1, \\ x_2(0) = 0, \end{cases} \quad 1 \le u \le 4 \quad ,$$

$$J = x_1(3\pi/4) \to \min .$$

В таблице приведены результаты численных экспериментов с использованием метода проекции градиента с точностью $\varepsilon = 0{,}00001$ при различных начальных значениях $\left(\nu_0, \tau_0\right)$ вектора параметров, $N = 20000$, $K = 2$. Минимальное (точное) значение функционала $J = -2$ достигается при $\left(\nu^1, \nu^2, \tau\right) = \left(4{,}1{,}0{,}78539\right)$.

Численные результаты решения задачи приведены в таблице

(ν,τ)	J	(ν,τ)	J^*	итераций
(3,200;1,500;1,231)	-1,2703	(4,00000 1,000000; 0,774556)	-1.99965	12
(2,850;1,200;0,522)	-1,346628	(4,00000 1,000000; 0,79229)	-1.99985	12
(3,150;2,430;0,953)	-0,852592	(4,00000 1,000000; 0,781399)	-1.999952	12
(3,540;1,820;1,847)	-0,44103	(4,00000 1,000000; 0,779683)	-1.999902	15
(2,180;1,470;1,368)	-1,1798	(4,00000 1,000000; 0,781340)	-1.999966	16
(1,890;0,750;2,092)	-1,02568	(4,00000 1,000000; 0,786724)	-1.999947	9

Заключение. Вид зависимости градиента целевой функции от параметров управления в классе кусочно-постоянных управлений позволяет применить прямой метод оптимизации градиентного типа с корректировкой шага для численного решения задач оптимального управления на классе кусочно-постоянных функций.

Численное решение задачи оптимального управления в классе кусочно-постоянных функций в связи с более простой технической реализацией управляющих приборов и устройств, использующих подобные управления, может быть использовано при разработке математического обеспечения систем автоматического управления на производстве в различных отраслях народного хозяйства.

Библиографический список

1. *Александров В.В., Бахвалов Н.С. и др.* Практикум по численным методам в задачах оптимального управления. М.: Изд-во Моск. ун-та, 1988. 80 с.

2. *Брайсон А., Хо Ю-Ши* Прикладная теория оптимального управления. М.: Мир, 1972. 545 с.

3. *Моисеев А.А.* Оптимальное управление при дискретных управляющих воздействиях //Автоматика и телемеханика. 1991. № 9. С. 123-132.

4. *Табак Д., Куо Б.* Оптимальное управление и математическое программирование. М.: Наука, 1975. 280 с.

5. *Федоренко Р.П.* Приближенное решение задач оптимального управления. М.: Наука, 1978. 486 с.

6. *Рагимов А.Б.* Об одном подходе к решению задач оптимального управления на классах кусочно-постоянных, кусочно-линейных, кусочно-заданных функций// Вестник ТГУ. 2012. № 2. Вып.19 С. 20-30.

7. *Кузнецов А.В.* Условия локальной оптимальности для нелинейных управляемых систем в классе кусочно-постоянных управлений// Вестник РГРТУ. 2011. №4. Вып. 38, С. 125128.