ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ЛИПЕЦКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

На правах рукописи

Щербаков Артем Петрович

РАЗРАБОТКА МЕТОДОВ И АЛГОРИТМОВ РЕКУРРЕНТНОЙ ИДЕНТИФИКАЦИИ ИЕРАРХИЧЕСКИХ ОКРЕСТНОСТНЫХ МОДЕЛЕЙ

Специальность 1.2.2 – Математическое моделирование, численные методы и комплексы программ

Диссертация на соискание ученой степени
кандидата технических наук

Научный руководитель:
доктор технических наук, профессор
Шмырин Анатолий Михайлович

Липецк – 2023

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ

Актуальность темы исследования и степень её разработанности. Многие современные сложные производственные процессы характеризуются распределенностью во времени и пространстве, нелинейностью и наличием спектра номинальных технологических режимов как производства в целом, так и отдельных его составляющих или узлов. Проявления нелинейности многообразны и, как правило, связаны со спецификой изучаемого процесса или объекта, и потому не существует универсальных методов построения нелинейных моделей. Разработка новых математических методов и алгоритмов, адекватно отражающих характер нелинейности отдельных номинальных режимов или всего моделируемого процесса в целом, остается актуальной задачей, несмотря на большое количество уже имеющихся методов и алгоритмов, даже таких мощных как нейронные сети или ансамблевые алгоритмы на основе деревьев регрессии.

Одним из перспективных направлений моделирования сложных распределенных нелинейных процессов является окрестностное моделирование, основанное на математической формализации технологической схемы процесса в виде окрестностной структуры – орграфа с наборами переменных и операторов, ассоциированных, соответственно, с вершинами и дугами орграфа. Построенной окрестностной структуре соответствует окрестностная система, уравнения которой, в зависимости от рассматриваемой задачи, могут быть статическими или динамическими, линейными или нелинейными и так далее. Теория окрестностных систем и связанные с ней задачи идентификации и управления рассматривались в работах С.Л. Блюмина, А.М. Шмырина, Н.Н. Карабутова, И.А. Седых и других авторов.

Нелинейность в окрестностных моделях связана, прежде всего, с выбором типа уравнений или операторов окрестностной системы. Во многих работах рассматривались билинейные и, в более общем случае,

полиномиальные окрестностные системы. Ограниченность области применения полиномиальных окрестностных систем связана с их локальностью (как и формулы Тейлора) и, кроме того, с проблемами мультиколлинеарности, возникающей в задаче идентификации в связи с введением искусственных предикторов, соответствующих нелинейным одночленам. В работах И.А. Седых рассматривались нейроокрестностные модели, в которых нелинейность в узлах (вершинах) моделировалась с помощью нейронных сетей как операторов узлов окрестностной системы, а нелинейность процесса в целом была связана с рассмотрением переменных окрестностей в динамическом случае. Для идентификации таких моделей, включающих нейронные сети, требуется большое количество данных. В диссертации А.С. Канюгиной рассматривалась схема объединения нескольких линейных окрестностных моделей, соответствующих номинальным технологическим режимам производства, в общую нелинейную модель с возможностью перехода между номинальными режимами.

Настоящая работа посвящена дальнейшему развитию методов нелинейного моделирования во взаимосвязи с теорией окрестностных систем. Уже в первых работах по окрестностному моделированию отмечалось, что окрестностные системы могут быть полезны не только в задачах моделирования производственных процессов, но и как средство интерпретации и анализа многих математических методов и алгоритмов, таких, например, как метод сеток, метод конечных элементов, нейронные сети и сети Петри. Продолжая данное «алгоритмическое» направление применения окрестностных систем, представленные в работе методы и алгоритмы рекуррентной иерархической идентификации нелинейных моделей описываются с помощью иерархических окрестностных структур и систем.

Рассматриваются методы уточнения заданной первоначальной модели (например, модели узлового оператора окрестностной системы) посредством рекуррентной идентификации дополнительных слагаемых на основании использования остаточных данных (невязок) предыдущей модели.

Корректирующие слагаемые соответствуют вершинам некоторой иерархической окрестностной структуры, орграф такой структуры называется деревом модели. В первом методе, который можно назвать статическим, дерево модели соответствует заранее заданной иерархической кластеризации множества кортежей входных данных, то есть дерево фиксировано и не изменяется. Во втором методе, который можно назвать динамическим, дерево модели соответствует возникающему в процессе идентификации иерархическому разбиению (фрагментации) множества входных кортежей, то есть дерево определяется в процессе идентификации одновременно с разбиением и не задано заранее. Важно отметить, что, в отличие, например, от нейросетевых методов, данные методы не требуют большого количества экспериментальных данных.

Наличие дерева в методах рекуррентной иерархической идентификации придает им сходство с известным алгоритмом CART (Classification and regression trees), разработанным Л. Брейманом, Д. Фридманом и другими, а рекуррентное использование остаточных данных – с ансамблевым методом бустинга на основе CART. В действительности представленные в работе алгоритмы, несмотря на указанное внешнее сходство, существенно отличаются от алгоритма CART и основанных на CART ансамблевых методах. Анализ сходства и различия алгоритмов удобно проводить, используя их представление в виде (определяемых в данной работе) иерархических окрестностных моделей.

Тематика работы связана с научными направлениями ФГБОУ ВО «Липецкий государственный технический университет» «Исследование и разработка методов и алгоритмов прикладной математики для идентификации технологических и сопровождающих процессов» и «Современные сложные системы управления».

Цель работы и задачи исследования. Целью диссертационной работы является разработка методов и алгоритмов рекуррентной идентификации иерархических окрестностных моделей и анализа остаточных данных и

применение разработанных алгоритмов в задачах моделирования процессов с нелинейным поведением. В рамках диссертационного исследования были поставлены и решены следующие задачи:

1. Обзор существующих классов окрестностных моделей и разработка моделей с иерархической окрестностной структурой.

2. Разработка алгоритмов и численных методов рекуррентной идентификации иерархических окрестностных моделей на основе использования остаточных данных выходов (невязок) промежуточных моделей.

3. Разработка метода контроля количества уровней в иерархических окрестностных моделях на основе анализа остатков промежуточных моделей.

4. Применение алгоритмов рекуррентной идентификации иерархических окрестностных моделей для исследования и прогнозирования свойств и характеристик процессов с нелинейным поведением.

5. Разработка проблемно-ориентированных модулей программного обеспечения, реализующих алгоритмы рекуррентной идентификации иерархических моделей.

Научная новизна. В диссертационной работе получены следующие результаты, отличающиеся научной новизной:

1. Введен класс окрестностных моделей, отличающийся иерархической структурой с двусторонними связями и возможностью добавления новых узлов, что позволяет расширить возможности применения окрестностных систем в задачах моделирования производственных процессов и задачах описания иерархических алгоритмов.

2. Разработан алгоритм рекуррентной идентификации иерархических окрестностных моделей на основе аппроксимации невязок промежуточных моделей, отличающийся использованием заданной иерархической кластеризации обучающих входных данных и позволяющий повысить точность моделей.

3. Разработан алгоритм и численный метод рекуррентной идентификации иерархических окрестностных моделей на основе трихотомии невязок промежуточных моделей, отличающийся построением в процессе идентификации иерархического разбиения обучающих входных данных и позволяющий улучшать точность моделей и качество прогноза.

4. Предложен метод анализа остаточных данных промежуточных моделей в алгоритмах рекуррентной идентификации, отличающийся введением порядка на множестве обучающих данных и позволяющий контролировать количество уровней иерархии, необходимое для достижения соответствия между точностью и качеством прогноза иерархической модели.

5. Разработан комплекс проблемно-ориентированных программ, реализующих алгоритмы рекуррентной идентификации иерархических моделей, отличающихся наличием модуля вычисления ближайшего обучающего кортежа и позволяющих оценивать длину дерева модели.

Теоретическая и практическая значимость.

Теоретическая значимость результатов работы заключается в разработанных алгоритмах рекуррентной идентификации иерархических окрестностных моделей аппроксимации данных.

Практическая значимость результатов работы заключается в построении моделей прогнозирования модульных характеристик клинкера после обжига в зависимости от химического состава до обжига и модели прогнозирования температурного режима стадии диффузии производства сахара.

Объект исследования. Производственные процессы с существенно нелинейными характеристиками.

Предмет исследования. Моделирование сложных производственных процессов с существенно нелинейными характеристиками с помощью алгоритмов рекуррентной идентификации иерархических моделей.

Методология и методы исследования. В качестве теоретической и методологической основы выступают методы теории математического

моделирования, вычислительной математики, теории графов, математической статистики.

Положения, выносимые на защиту:

1. Новый класс окрестностных моделей с иерархической структурой, позволяющий расширить возможности применения окрестностных систем в задачах описания иерархических алгоритмов и моделирования нелинейных процессов.

2. Алгоритм рекуррентной идентификации иерархических окрестностных моделей на основе аппроксимации невязок промежуточных моделей, отличающийся использованием заданной иерархической кластеризации обучающих входных данных.

3. Алгоритм и численный метод рекуррентной идентификации иерархических окрестностных моделей на основе трихотомии невязок промежуточных моделей, отличающийся построением в процессе идентификации иерархического разбиения обучающих входных данных.

4. Метод контроля количества уровней иерархии в моделях, необходимых для достижения соответствия между точностью и качеством прогноза иерархической модели.

5. Комплекс проблемно-ориентированных программ, реализующих алгоритмы рекуррентной идентификации иерархических моделей и анализа остаточных данных.

Тематика работы соответствует следующим пунктам паспорта специальности 1.2.2:

разработка алгоритма иерархической идентификации на основе аппроксимации невязок промежуточных моделей, разработка алгоритма и численного метода иерархической идентификации на основе трихотомии невязок промежуточных моделей, позволяющих повысить качество аппроксимации и точность прогноза соответствует пункту 2 «Разработка, обоснование и тестирование эффективных вычислительных методов с применением современных компьютерных технологий»;

разработка метода проверки адекватности иерархических моделей, отличающегося анализом остаточных данных промежуточных моделей, упорядоченных по возрастанию обучающих выходов, и позволяющий контролировать количество необходимых уровней иерархии в моделях соответствует пункту 5 «Разработка новых математических методов и алгоритмов валидации математических моделей объектов на основе данных натурного эксперимента или на основе анализа математических моделей»;

исследование окрестностных моделей с иерархической структурой прогнозирования свойств клинкера в цементном производстве и температурного режима стадии диффузии производства сахара соответствует пункту 8 «Комплексные исследования научных и технических проблем с применением современной технологии математического моделирования и вычислительного эксперимента».

Степень достоверности результатов работы. Достоверность результатов работы подтверждается проведёнными в достаточном объеме исследованиями с применением современных технологий математического моделирования и вычислительного эксперимента. Разработанные методы, процедуры и алгоритмы были применены для исследования реальных объектов

– химического состава сырьевой муки и клинкера при обжиге клинкера в печах;

– температурных показателей процесса диффузии и экстрагирования производства сахара.

Сравнительный анализ полученных результатов показал соответствие производственным данным.

Реализация и внедрение результатов работы. Результаты диссертационной работы рекомендованы к использованию акционерным обществом «Липецкцемент» для анализа и управления технологическими показателями предприятия; рекомендованы к использованию на структурном подразделении «Хмелинецкий сахарный завод» акционерного общества

«Агропромышленное объединение «Аврора» для уточнения и совершенствования существующих математических моделей сложных производственных процессов.

Теоретические результаты диссертации используются в учебном процессе ФГБОУ ВО «Липецкий государственный технический университет» при выполнении курсовых работ по дисциплине «Математическое моделирование», а также подготовке выпускных квалификационных работ.

Апробация работы. Теоретические и практические результаты работы докладывались и обсуждались на международных конференциях: Международная научно-техническая конференция «Фундаментальные и прикладные проблемы модернизации современного машиностроения и металлургии» (Липецк, 2012), XVII Международная научная конференция «Современные проблемы информатизации в экономике и обеспечении безопасности» (Воронеж, 2012), Международная научно-практическая конференция «Современные проблемы в образовании и науке» (Тамбов, 2013), VI международная конференция «Современные методы прикладной математики, теории управления и компьютерных технологий» («ПМТУКТ») (Воронеж, 2013), XX-th International Open Science Conference «Modern informatization problems in simulation and social technologies» (Yelm, WA, USA, 2015), 2nd International Conference on Technology Enhanced Learning in Higher Education TELE2022 (Lipetsk, 2022), 4rd International Conference on Control Systems, Mathematical Modeling, Automation and Energy Efficiency SUMMA2022 (Lipetsk, 2022), 5th International Conference on Control Systems, Mathematical Modeling, Automation and Energy Efficiency SUMMA2023 (Lipetsk, 2023), Международная научно-практическая конференция «Нано-био-технологии. Теплоэнергетика. Математическое моделирование» (Липецк, 2023); всероссийских конференциях: IX всероссийская школа-конференция молодых ученых «Управление большими системами» (Тамбов, 2012), областных региональных конференциях, а также на научных семинарах

кафедры высшей математики Липецкого государственного технического университета.

Публикации. Основные результаты работы опубликованы в 26 печатных трудах, в том числе 5 – самостоятельно, из них 5 статей в ведущих рецензируемых журналах, рекомендованных в Перечне ВАК, 2 – в изданиях, входящих в международные системы цитирования Scopus и Web of Science, 3 свидетельства о государственной регистрации программы для ЭВМ.

В работах, написанных в соавторстве и приведенных в автореферате, лично соискателем получены следующие результаты: [2, 12] – разработка и применение на сгенерированных данных алгоритмов рекуррентной идентификации иерархических окрестностных моделей, определение иерархического разбиения и иерархической кластеризации, построение непрерывной модели с помощью иерархического разбиения единицы; [3, 7] – разработка схемы агрегирования локальных мод системы, зависящих от кластеров входов, в общую мультимодальную окрестностную систему; [4, 15-17, 20] – определение компонент состояния и управления окрестностной системы печи обжига клинкера, [5, 21, 23] – оценка адекватности общих и параметрических окрестностных систем; [6] – разработка методов стабилизации локальных мод; [8] – разработка программного обеспечения, реализующего численные методы анализа нелинейности входного процесса; [9] – алгоритмическое обеспечение для получения общего параметрического уравнения окрестностной модели; [10] – алгоритмическое обеспечение для решения систем методом ортонормализации Грама-Шмидта; [19] – анализ нелинейности при исследовании характеристик производства клинкера; [22, 24-26] – разработка окрестностных моделей распределенных динамических систем.

Структура и объем работы. Диссертация состоит из введения, четырёх глав, заключения, списка литературы из 134 наименований и 2 приложений на 6 страницах. Объем основной части работы составляет 134 страницы, включая 24 рисунка и 6 таблиц.

Во введении обоснована актуальность темы исследования, сформулированы цели и задачи работы, научная новизна, теоретическая и практическая значимость работы, методология и методы исследования, перечислены положения, выносимые на защиту, степень достоверности и апробации результатов, а также основное содержание работы.

В первой главе проведен обзор различных видов линейных и нелинейных окрестностных моделей, в том числе описанных в литературе классов окрестностных моделей иерархического типа. Указаны два направления применения окрестностных моделей, технологическое и вычислительное. Представлены сведения об алгоритмах решающих деревьев (алгоритмы CART) и основанных на CART ансамблевых методах. Обсуждается проблема соотношения точности моделей и качества прогноза.

Вопросами идентификации линейных и нелинейных окрестностных систем занимались С.Л. Блюмин, А.М. Шмырин, Н.Н. Карабутов, И.А. Седых и другие авторы. В работах А.М. Шмырина и И.А. Седых были определены два класса иерархических окрестностных моделей. Также отмечается два направления применения окрестных моделей – «технологическое» и «вычислительное».

Ставится задача разработки нового класса окрестностных моделей, имеющих иерархическую окрестностную структуру, и алгоритмов иерархической идентификации окрестностных моделей с использованием методов анализа остаточных данных для контроля количества уровней иерархии.

Во второй главе определяются иерархические окрестностные модели с окрестностной структурой в виде ориентированного дерева с восходящими и нисходящими дугами или дугами только одного типа. Этот класс иерархических окрестностных моделей не совпадает с ранее определенными классами, описанными в первой главе. В данной работе эти модели используются для представления разработанных алгоритмов рекуррентной идентификации и для анализа сходства и различия этих алгоритмов с

алгоритмом CART и ансамблевым методом бустинга на основе CART. Далее термин «иерархическая окрестностная модель» используется для моделей с древовидной окрестностной структурой.

Описываются иерархические кластеризации и иерархические разбиения обучающих данных, определяются связанные с кластеризациями и разбиениями иерархические окрестностные структуры. Дается интерпретация процесса построения и дальнейшего использования дерева регрессии в алгоритме решающих деревьев (CART) как процесса идентификации и последующего применения иерархической окрестностной модели с нисходящими связями.

В третьей главе описываются статический и динамический алгоритмы рекуррентной идентификации иерархических окрестностных моделей. Предлагается схема контроля момента остановки рекуррентных алгоритмов при достижении соответствия точности и качества прогноза модели на основе анализа остаточных данных выходов уже построенных моделей для данного уровня иерархии. Рассматривается задача синтеза мультимодальных окрестностных систем на основе агрегирования номинальных режимов системы в целом или номинальных режимов отдельных узлов.

Оба алгоритма рекуррентной идентификации иерархических моделей представляют собой схему последовательного уточнения некоторой начальной аналитической модели, при этом окончательный результат интерпретируется как иерархическая окрестностная модель с восходящими дугами. Окрестностная структура иерархической модели (дерево модели) в статическом алгоритме известна заранее и соответствует заданной иерархической кластеризации (или разбиению) множества обучающих данных. В динамическом алгоритме окрестностная структура заранее не известна и определяется в процессе последовательного уточнения модели одновременно с построением иерархического разбиения множества обучающих данных.

Также рассматривается метод анализа исходных данных и остаточных данных иерархических моделей. Метод анализа остаточных данных необходим для контроля количества уровней в иерархических моделях, основан на использовании коррелограммы автокорреляционной функции (АКФ) остатков модели, распределенных в соответствии с упорядоченными значениями выходной переменной. С увеличением уровня иерархии (ростом длины дерева *T*) величина автокорреляций снижается. Такая ситуация означает, что иерархические модели с добавлением новых уровней точнее аппроксимируют моделируемый процесс, оставляя в остатках все меньше информации. Отсутствие какой-либо информации о процессе в остатках модели очередного уровня эквивалентно ситуации, когда АКФ для остатков мало отличима от АКФ для белого шума. Дальнейшее уточнение иерархической модели в этом случае возможно, но не имеет смысла, так как приводит к получению переобученной модели.

Предлагаются две схемы построения мультимодальной окрестностной системы: на основе идентификации локальных мод системы в целом (с последующим агрегированием этих мод в общую систему) и на основе агрегирования микролокальных мод узлов, без идентификации локальных мод всей системы.

В четвёртой главе рассматриваются примеры построения иерархических моделей с помощью статического и динамического алгоритмов рекуррентной идентификации с анализом остаточных данных, а также применение алгоритмов к задачам построения иерархических окрестностных моделей прогнозирования модульных характеристик клинкера после обжига и прогнозирования температурных режимов стадии диффузии производства сахара.

Представлено сравнение разработанных иерархических моделей, идентифицированных посредством L-алгоритма (с кластеризацией и трихотомией), с традиционными линейной и квадратичной регрессионными моделями, моделями нейронных сетей и ансамблевыми моделями деревьев

регрессии, такие как бустинг и случайный лес, в задачах прогнозирования модульных характеристик клинкера и температурных режимов стадии диффузии производства сахара. Представлены оценки различных показателей качества аппроксимации для всех рассматриваемых моделей, сделаны выводы.

В заключении изложены основные результаты исследования, представлены выводы и рекомендации, перспективы дальнейших разработок по данной теме.

1. ОБЗОР МЕТОДОВ ОКРЕСТНОСТНОГО МОДЕЛИРОВАНИЯ

В данной главе представлен обзор теории и приложений окрестностных моделей. Рассмотрены линейные и нелинейные окрестностные модели, в том числе введенные ранее в литературе окрестностные модели, определяемые как иерархические. Описаны два направления применения окрестностных моделей, приведены примеры соответствующих приложений. Рассмотрены вопросы идентификации моделей и оценки соответствия между точностью и качеством прогноза моделей. Приведен обзор различных вариантов решения проблемы улучшения прогностических свойств моделей.

1.1. Окрестностные структуры и системы

Окрестностные системы являются логическим продолжением работ в области дискретных систем. Вводимые в работах [5-6, 13-15, 17, 34-36, 74, 80, 83, 92-93] окрестностные системы и модели являются обобщающими для целого ряда дискретных моделей, распределенных и сосредоточенных, пространственных, временных и пространственно-временных, линейных и нелинейных, четких и нечетких.

Задаче управления каким-либо процессом в большинстве случаев предшествует задача идентификации модели этого процесса. Идентификацию окрестностных систем, как правило, можно разбить на два этапа – структурную идентификацию и параметрическую идентификацию (см. [4, 6, 9, 13, 15]). Структурная идентификация подразумевает задание узлов модели, связей между ними и соответствующих наборов переменных (входы, состояния, выходы). Далее определяются уравнения, связывающие входы, состояния и выходы, и задаются неизвестные параметры уравнений, которые на втором этапе подлежат параметрической идентификации.

При структурной идентификации задается окрестностная структура (см. [46-48]) в виде ориентированного графа $\Re = (\hat{V}; E)$, с помощью которого

удобно рассматривать моделируемый процесс с его составными частями. Орграф окрестностной структуры является неполносвязным: могут быть вершины без петель, а также вершины без входящих ребер (дуг) и вершины без выходящих ребер. Такие вершины при введении окрестностных структур соответствуют входам и выходам дискретной системы. Вершины, имеющие как входящие, так и выходящие дуги называются узлами. Дуги окрестностной структуры авторы [46-48] называют также связями. Объединение входов U, узлов V и выходов W образует конечное множество всех вершин $\hat{V} = U \cup V \cup W$, а множество E – множество дуг орграфа. При этом:

1) каждая вершина инцидентна, по крайней мере, одной дуге;

2) входы $u \in U$ (выходы $w \in W$) связаны с другими вершинами только выходящими $e(u, *)$ (входящими $e(*, w)$) дугами, где символ $*$ обозначает произвольную вершину орграфа;

3) любые два узла орграфа $v', v'' \in V$ либо не являются смежными (не имеют связи друг с другом), либо имеют связь в виде одной дуги или двух противоположно направленных дуг $e(v', v'')$ и $e(v'', v')$;

4) узлы $v \in V$ могут иметь петли, т.е. дуги вида $e(v, v)$, такие узлы называются рефлексивными;

5) каждый узел $v \in V$ имеет входящие и выходящие дуги.

Если все узлы окрестностной структуры рефлексивны, то и сама структура называется рефлексивной. Источниками вершины $\hat{v}$ называются все вершины, имеющие входящую связь для данной вершины $\hat{v}$, и стоками – все вершины, связанные с $\hat{v}$ исходящими дугами. Как минимум одним стоком и источником обладает каждый узел (вершина из V) окрестностной структуры. Рефлексивные узлы являются сами для себя стоками и источниками. При этом все входы $u \in U$ имеют только стоки, а все выходы $w \in W$ – только источники, таким образом, ни входы, ни выходы не являются рефлексивными.

Пример окрестностной структуры изображен на рисунке 1.1. На нем вершины 1 и 2 – это входы, вершины 8, 9, 10 – выходы, вершины 3, 4, 5, 6, 7 – узлы, узлы 4, 5 и 6 являются рефлексивными.

Рефлексивными входом и выходом называются узлы, соответственно, без входящих простых дуг и без выходящих простых дуг, имеющих петли. На рисунке 1.1 окрестностная структура не имеет рефлексивных входов, но имеет один рефлексивный выход – вершина 5.

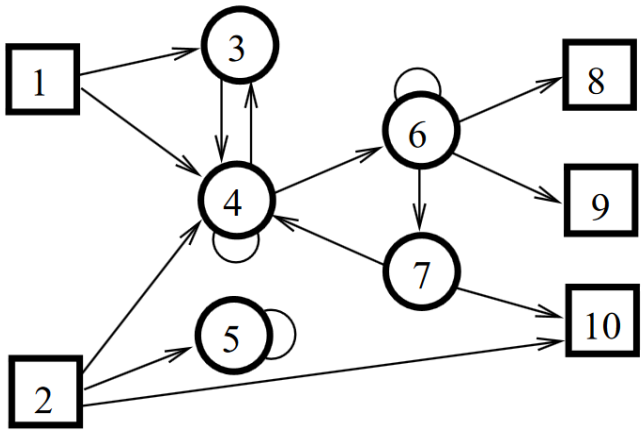


Рисунок 1.1 – Пример окрестностной структуры

При построении окрестностной структуры отмечается необязательность связности орграфа, т.е. в частных случаях могут иметь место изолированные рефлексивные узлы. Такие структуры встречаются, например, при описании некоторых видов иерархических окрестностных структур (см. [56, 59-60, 89, 96]).

1.2. Линейные, нелинейные и иерархические окрестностные модели

1.2.1. Линейные и нелинейные окрестностные модели

В работах [4-5, 7, 9, 13, 16-17, 34-36, 86, 92-95, 99, 108] были введены и рассмотрены линейные (симметричные и смешанные), нелинейные (билинейные одноаргументные и m-линейные (n_1 +…+ n_m)-аргументные) окрестностные системы, а также системы с нечеткими окрестностями и с динамическими окрестностями.

Окрестностная модель в общем случае описывается набором (см. [86, 94-95, 99]):

$$NS = (N, X, V, Y, Z, G, F, X[0]), \qquad (1.1)$$

где 1) $N = (A, O_v, O_x, O_y)$ – структура окрестностной модели, $A = \{a_1, a_2, ..., a_N\}$ – множество узлов, O_v – окрестности связей узлов по управлению, O_x – окрестности связей узлов по состоянию, O_y – окрестности связей узлов по выходным воздействиям. Для каждого узла $a_i \in A$ ($i = 1,\ldots,N$) определена своя окрестность по состояниям $O_x[a_i] \in A$, управлениям $O_v[a_i] \in A$ и выходам $O_y[a_i] \in A$;

2) $X \in R^n$ – вектор состояний окрестностной модели в текущий момент времени; $V \in R^m$ – вектор управлений (входов) окрестностной модели в текущий момент времени; $Y \in R^l$ – вектор выходов окрестностной модели в текущий момент времени; $Z \in R_+^n$ – вектор переменных задержек в узлах, где R_+ – множество неотрицательных действительных чисел;

3) $G : X_{O_x} \times V_{O_v} \to X$ – функция пересчета состояний окрестностной модели, где X_{O_x} – множество состояний узлов, входящих в окрестность O_x, V_{O_v} – множество управлений узлов, входящих в окрестность O_v;

4) $F : X_{O_x} \times V_{O_v} \to Y$ – функция пересчета выходов окрестностной модели;

5) $X[0]$ – начальное состояние модели.

В частных задачах некоторые из составляющих окрестностной модели могут отсутствовать.

Функции пересчета состояний *G* и выходов *F* в общем случае могут быть недетерминированными. В частных случаях данные функции могут быть линейными, билинейными, полиномиальными.

Простейшими линейными окрестностными моделями являются симметричная (учитывающая состояние и вход) и смешанная (учитывающая состояние, вход и выход) модели (см. [13, 17]):

$$\sum_{\alpha\in O_x[a]} w_x[a,\alpha]X[\alpha] = \sum_{\beta\in O_v[a]} w_v[a,\beta]V[\beta] \tag{1.2}$$

$$\sum_{\alpha\in O_x[a]} w_x[a,\alpha]X[\alpha] + \sum_{\beta\in O_v[a]} w_v[a,\beta]V[\beta] + \sum_{\gamma\in O_y[a]} w_y[a,\gamma]Y[\gamma] = 0 \tag{1.3}$$

где $X[a] \in R^n$, $V[a] \in R^m$, $Y[a] \in R^q$ – векторы состояния, входа (управления) и выхода узла a системы; $w_x[a,\ \alpha] \in R^{c\times n}$, $w_v[a,\ \beta] \in R^{c\times m}$, $w_y[a,\ \gamma] \in R^{c\times q}$ – матрицы, подлежащие параметрической идентификации; $O_x[a]$, $O_v[a]$, $O_y[a]$ – окрестности узла a, соответственно, по состоянию, входу и выходу; $a,\alpha,\beta,\gamma \in A$, $A = \{a_1, a_2, \dots, a_N\}$ – конечное множество узлов системы, $|A| = N$.

Модели (1.2) и (1.3) в общем виде представляются:

$$\Phi_x(x[O_x(a)]) = \Phi_v(v[O_v(a)]); \tag{1.4}$$

$$\mathrm{F}(\Phi_x(x[O_x(a)]),\Phi_v(v[O_v(a)]),\Phi_y(y[O_y(a)])) = 0, \tag{1.5}$$

Среди нелинейных окрестностных систем наиболее простым классом являются, рассмотренные в [5, 17], билинейные окрестностные модели, учитывающие влияние входного воздействия и состояния:

$$\begin{aligned}&\sum_{\alpha\in O_x[a]} w_x[a,\alpha]X[\alpha] + \sum_{\beta\in O_v[a]} w_v[a,\beta]V[\beta] + \\ &+ \sum_{\alpha\in O_x[a]}\sum_{\gamma\in O_y[a]} [w_{1vx}[a,\alpha,\beta]v[\beta,1]X[\alpha] + \dots + w_{mvx}[a,\alpha,\beta]v[\beta,m]X[\alpha]] = 0,\end{aligned} \tag{1.6}$$

где $w_{ivx}[a,\ \alpha] \in R^{c\times n}$ $(i = 1, \dots, m)$ – матрицы-параметры; $v[a,\ i]$ – элементы кортежей входов $V[a]$.

Билинейная модель (1.6) в короткой записи:

$$\sum_{\alpha\in O_x[a]} w_x[a,\alpha]X[\alpha] + \sum_{\beta\in O_v[a]} w_v[a,\beta]V[\beta] + \sum_{\substack{\alpha\in O_x[a]\\ \beta\in O_v[a]}} w_{xv}[a,\alpha,\beta]X[\alpha]V[\beta] = 0, \quad (1.7)$$

где $w_{xv}[a,\ \alpha,\ \beta] \in R^{c\times n\times m}$ – блочные матрицы-параметры.

Например, в случае одного узла уравнение билинейной окрестностной модели (1.6)–(1.7) будет выглядеть:

$$w_x[1,1]x[1] + w_v[1,1]v[1] + w_{xv}[1,1,1]x[1]v[1] = 0 \quad (1.8)$$

В случае двух узлов билинейная модель имеет вид:

$$\begin{aligned} &w_x[1,1]x[1] + w_x[1,2]x[2] + w_v[1,1]v[1] + w_v[1,2]v[2] + \\ &+w_{xv}[1,1,1]x[1]v[1] + w_{xv}[1,1,2]x[1]v[2] + w_{xv}[1,2,1]x[2]v[1] + \\ &+w_{xv}[1,2,2]x[2]v[2] = 0 \\ &w_x[2,1]x[1] + w_x[2,2]x[2] + w_v[2,1]v[1] + w_v[2,2]v[2] + \\ &+w_{xv}[2,1,1]x[1]v[1] + w_{xv}[2,1,2]x[1]v[2] + w_{xv}[2,2,1]x[2]v[1] + \\ &+w_{xv}[2,2,2]x[2]v[2] = 0 \end{aligned} \quad (1.9)$$

И так далее для большего количества узлов окрестностной системы.

Нелинейная окрестностная модель в общем случае записывается в виде (см. [13, 79]):

$$\Phi(a; \{X(\alpha), \alpha \in O_x[a]\}; \{V(\beta), \beta \in O_v[a]\}; \{Y(\gamma), \gamma \in O_y[a]\}) = 0 \quad (1.10)$$

где $a \in A$; $O_v[a]$, $O_x[a]$, $O_y[a]$ – окрестности узла a системы по входу v, состоянию x и выходу y соответственно; $a, \alpha, \beta, \gamma \in A$.

Продолжением развития теории окрестностного моделирования является исследование окрестностных моделей, учитывающих степень нечёткого взаимного влияния узлов, входящих в окрестности по входам, состояниям и выходам (см. [10, 16, 34, 58, 74, 77, 93]). Например, при нечетком задании окрестностей симметричной системы (1.2), окрестности вершины (узла) по состоянию и входному воздействию расширяются до всего множества A, что соответствует появлению весов в каждом слагаемом:

$$\sum_{\alpha\in A} w_x[a,\alpha]M_x[a,\alpha]X[\alpha] = \sum_{\beta\in A} w_v[a,\beta]N_v[a,\beta]V[\beta] \quad (1.11)$$

где $M_x[a,\alpha] \in R^{n\times k}, N_v[a,\beta] \in R^{n\times l}$ – матрицы коэффициентов, элементы которых $m_{ij}, n_{fg} \in [0,\ 1]$.

Нелинейная смешанная нечётко-окрестностная модель описывается уравнением (см. [16, 92-93]):

$$\begin{aligned}&\Phi(a;\{\mu_x, x(\alpha), \alpha \in O_x[a]\};\{\mu_v, v(\beta), \beta \in O_v[a]\};\\&\{\mu_y, y(\gamma), \gamma \in O_y[a]\}) = 0\end{aligned} \tag{1.12}$$

где $\mu_x,\ \mu_v,\ \mu_y \in [0,\ 1]$ – функции принадлежности, являющиеся элементами матриц инцидентности, соответственно, по входу, состоянию и выходу. Они характеризуют степень нечеткого влияния друг на друга элементов окрестностей $O_v,\ O_x,\ O_y$.

Рассмотренные выше окрестностные модели являются статическими, что ограничивает их применимость для моделирования объектов без учета времени. В дальнейшем, например в [5, 129], вводится динамика в окрестностные модели. Например, в случае линейности *G* и *F* окрестностно-временная модель представляет собой систему линейных уравнений (см. [57, 65, 99]):

$$\begin{cases}\sum\limits_{\alpha\in O_x[t+1,a_i]} w_x[t+1,a_i,\alpha]x[t+1,\alpha] = \\ = \sum\limits_{\alpha\in O_x[t,a_i]} w_x[t,a_i,\alpha]x[t,\alpha] + \sum\limits_{\beta\in O_v[t,a_i]} w_v[t,a_i,\beta]v[t,\beta] \\ \sum\limits_{\gamma\in O_y[t+1,a_i]} w_y[t+1,a_i,\gamma]y[t+1,\gamma] = \\ = \sum\limits_{\alpha\in O_x[t,a_i]} w_x[t,a_i,\alpha]x[t,\alpha] + \sum\limits_{\beta\in O_v[t,a_i]} w_v[t,a_i,\beta]v[t,\beta]\end{cases}, \tag{1.13}$$

где $x[t,a_i] \in R^n, v[t,a_i] \in R^m$ – состояние и вход в узле a_i в момент времени *t*; $x[t+1,a_i] \in R^n, y[t+1,a_i] \in R^l$ – состояние и выход в узле a_i в момент времени $t + 1$.

Модель (1.13) в матричном виде имеет вид:

$$\begin{cases} W_x[t+1]X[t+1] = W_x[t]X[t] + W_v[t]V[t] \\ W_y[t+1]Y[t+1] = W_x[t]X[t] + W_v[t]V[t]. \end{cases} \quad (1.14)$$

В случае нелинейности функций G и F окрестностную модель с динамикой в общем виде можно представить:

$$\begin{cases} W_x[t+1]X[t+1] = G(X[t], V[t]) \\ W_y[t+1]Y[t+1] = F(X[t], V[t]). \end{cases} \quad (1.15)$$

1.2.2. Иерархические структуры и иерархические окрестностные модели

В работах А.М. Шмырина и И.А. Седых были определены два класса иерархических окрестностных моделей (см. [56, 58-60, 89, 96]). В первом случае иерархичность модели понималась как многослойность окрестностной структуры с направлением связей от старших уровней иерархии (слоев) к младшим и произвольными связями внутри слоев, то есть каждый отдельно рассматриваемый слой мог быть произвольной окрестностной моделью.

Общая модель, отражающая влияние как окрестностей отдельных узлов структуры, так и уровней иерархии окрестностной системы, имеет вид:

$$\sum_{i=1}^{r} \sum_{\alpha \in O_{u_i}[a^{(1)}]} w_i[a^{(1)}, \alpha] u_i[\alpha] +$$

$$+ \sum_{i=1}^{r} \sum_{\alpha \in O_{u_i}[a^{(2)}]} \sum_{\beta \in O_{\gamma_i}[a^{(2)}]} w_i[a^{(2)}, \alpha, \beta] u_i[\alpha] \gamma_i[\beta] + \ldots \quad (1.16)$$

$$\ldots + \sum_{i=1}^{r} \sum_{\alpha \in O_{u_i}[a^{(k)}]} \ldots \sum_{\beta \in O_{\gamma_i}[a^{(k)}]} w_i[a^{(k)}, \alpha, \ldots, \beta] u_i[\alpha] \ldots \gamma_i[\beta] + \ldots = 0.$$

Здесь $w_i[a, \alpha_1, \ldots, \alpha_{n_1}, \alpha_{n_1+\ldots+n_{m-1}+1}, \ldots, \alpha_{n_1+\ldots+n_m}] \in \mathbb{R}^{c \times n_1 \times \ldots \times (n_1+\ldots+n_m)}$, $u_i[\alpha] \in \mathbb{R}^{n_i}, \ldots, \gamma_i[\beta] \in \mathbb{R}^{m_i}, \ldots$, $O_{u_i}[a^{(k)}], O_{\gamma_i}[a^{(k)}]$ – окрестности по сигналам u_i, γ_i $(i = 1, \ldots, r)$ узлов $a^{(k)}$, принадлежащих k-му слою $(k = 1, \ldots, L_k)$, c – количество строк матрицы w, $a \in A = \{a_1, \ldots, a_N\}$.

Иерархическая окрестностная структура, заданная иным образом, в общем случае имеет иное количество сигналов u_i, γ_i. Например, увеличение слоев иерархии в структуре приведет к появлению новых сигналов.

Во втором случае иерархичность модели понималась как введение дополнительных окрестностных моделей второго уровня внутри узлов первого уровня, т.е. узлы первого (внешнего) уровня структуры могут иметь определенную окрестностную структуру второго (внутреннего) уровня и описываться окрестностными моделями.

Различные иерархические структуры, не являющиеся окрестностными, рассматривались многими зарубежными и отечественным авторами. В частности, в работах Д.А. Новикова, А.А. Воронина, М.В. Губко, С.П. Мишина авторы рассматривают задачи оптимизации иерархических структур, в основном, применительно к организационным системам (см. [22, 23, 25]). Задача организационного дизайна включает в себя, в том числе, поиск оптимальной иерархической структуры организации/управления организацией.

В данных работах предлагаются модели иерархических организационных структур, основанные, например, на задании и поиске критерия эффективности, позволяющего сравнивать иерархические структуры и выбирать среди них оптимальную. В иерархиях управления естественным образом возникают подчинения и воздействия от менеджеров к исполнителям. Таким образом, иерархические организационные структуры можно рассматривать как ориентированные графы $H =< V, E >$ с множеством вершин V и множеством дуг $E \subseteq V \times V$. Если рассматривать одностороннюю связь (только управление), то граф структуры должен быть ацикличным, т.е. нельзя идя по дугам, вернуться в исходную вершину. Иными словами, исполнители не могут управлять менеджерами, а менеджеры более низких иерархий не могут управлять менеджерами более высоких иерархий (нет дуг, идущих в противоположную сторону при односторонних связях).

Если среди множества сотрудников организации (менеджеров и исполнителей) существует единственный менеджер, не имеющий выше себя в иерархии начальников (топ-менеджер), и при этом каждый сотрудник, кроме топ-менеджера имеет только одного непосредственного начальника, то такая иерархическая структура H называется деревом. Примеры деревьев управления организацией представлены на рисунке 1.2, а-в. Исполнители при такой структуре соответствуют листьям дерева, топ-менеджер – корню. Если в организации есть менеджеры промежуточных уровней между топ-менеджером и исполнителями, то в структуре дерева существуют узлы, не являющиеся листьями и корнем (рисунок 1.2, а, в), при этом видно, что структура дерева может являться, например, классическим бинарным деревом. В случае только одного менеджера, управляющего всеми исполнителями, иерархия называется веерной (рисунок 1.2, б).

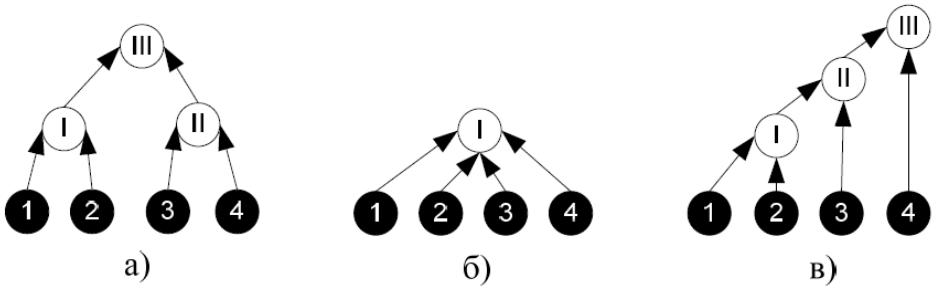


Рисунок 1.2 – Примеры иерархических структур управления в виде деревьев

Задача поиска оптимальной иерархии организации (в частности, поиска оптимального дерева) авторы формулируют так: для заданного множества исполнителей N на множестве всех допустимых иерархий $\Omega = \Omega(N)$ задается функционал $P: \Omega \to [0; +\infty)$. Оптимальной будет считаться иерархия (в частности, дерево) H^*, обеспечивающая минимум выбранного функционала:

$$H^* \in \underset{H \in \Omega}{\operatorname{Arg\,min}} P(H). \qquad (1.17)$$

В качестве функционала $P(H)$ может выступать, например, неотрицательная функция затрат иерархии. При выборе иерархической

структуры, минимальные затраты иерархии подразумевают оптимальность этой иерархии управления.

1.3. Окрестностные модели в приложениях

Можно выделить два направления применения окрестностных моделей: технологическое, связанное с моделирования производственных процессов, и алгоритмическое, связанное с интерпретацией и анализом некоторых математических методов и алгоритмов, таких, например, как метод сеток, метод конечных элементов, нейронные сети и сети Петри. Приведем примеры соответствующих приложений.

В рамках первого направления приводятся результаты окрестностной формализации технологических схем. Например, в работе [5] рассматривается окрестностная модель цеха очистки сточных вод. В качестве узлов окрестностной структуры авторы выбирают 4 основных подсистемы, у которых можно измерить существенные параметры состояния, входа и выхода, – «вход в систему (на очистные сооружения)», «состояние после усреднения», «состояние после механической очистки», «сброс в реку», а также вводят дополнительный узел в структуру «цех в целом», позволяющий доопределить существенные параметры и качественнее идентифицировать модель и дальнейшее управление. Орграф рассматриваемого процесса представлен на рисунке 1.3.

В работах [101, 102] рассматривается параметрическая билинейная окрестностная модель системы теплоснабжения, в которой узлам системы соответствуют 3 подсистемы – «теплоэлектроцентраль», «насос» и «многоквартирный дом». Входами и выходами подсистем являются наиболее существенные параметры, например, площадь отапливаемого дома, подача насоса и прочие.

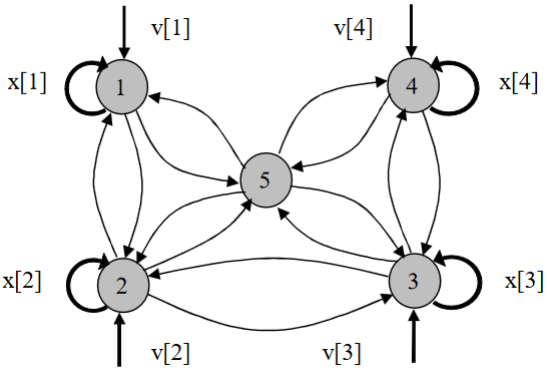


Рисунок 1.3 – Орграф цеха очистки сточных вод

В работах [91, 128] окрестностный подход применяется для моделирования установки поддержания оптимальной температуры полиола. Пять узлов окрестностной структуры у авторов соответствуют таким подсистемам, как «емкость для хранения полиола», «насос для перекачки полиола», «теплообменный аппарат», «потребитель полиола», «холодильная машина».

В работах [78, 104, 130] рассматривается окрестностное моделирование печей обжига клинкера. В работе печь условно разделяют на 4 зоны, соответствующих узлам билинейной окрестностной модели – «зона подогрева сырья», «зона декарбонизации», «зона появления клинкера», «зона охлаждения». Граф окрестностной системы представлен на рисунке 1.4.

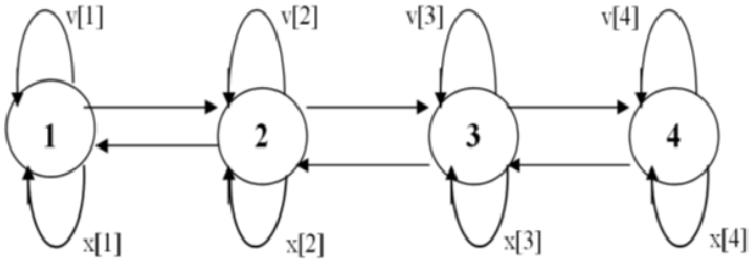


Рисунок 1.4 – Орграф зон вращающейся печи

В данных работах также применяется методика параметрического представления окрестностных переменных состояния $X[a] \in R^n$ и входа

$V[a] \in R^m$, нашедшая отражение и в других работах (см, например, [84, 85, 90]), а также разработанной программы для ЭВМ, предназначенной для определения общего параметрического уравнения каждого узла окрестностной модели в отдельности и всей модели в целом (см. [120]).

В работах [62, 72, 103, 110] окрестностные системы и структуры применяются для моделирования вентиляции и фильтрации производственных помещений, в частности, цехов цементного производства. Авторы, в общем случае, выделяют 4 узла системы – «приточный воздух», «фильтрация и терморегуляция», «производственных цех», «вытяжка и фильтрация воздуха».

В работах [59-60] автором рассмотрены возможности применения окрестностного моделирования уровня подземных вод месторождения цементного сырья. Узлам графа окрестностной системы, представленной на рисунке 1.5, соответствуют подсистема «внешняя среда» и 7 скважин. Также в данных работах, как уже отмечалось в пункте 1.2.2, рассматривается иерархичность окрестностной структуры в следующем понимании – некоторые узлы первого уровня («внешняя среда») задаются как окрестностные системы, т.е. имеют определенную окрестностную структуру второго уровня («внутренняя среда»).

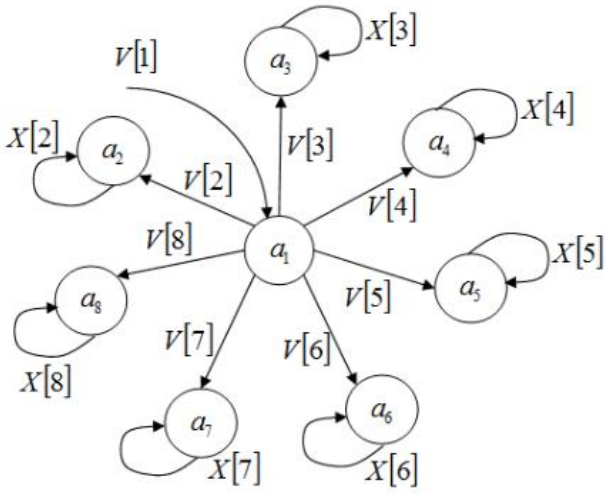


Рисунок 1.5 – Граф внешней структуры окрестностной модели уровня подземных вод

В рамках второго направления окрестностные модели рассматриваются в качестве некоторых известных вычислительных методов и алгоритмов, окрестностная структура которых не сопоставляется какому-либо производственному процессу или производственной системе. Например, в работах [6, 9, 13] окрестностный подход применяется к описанию таких сосредоточенных систем, для которых аргументом является время, в случае конечных алфавитов, трактуемых как конечные автоматы, и имеющее вид:

$$\begin{aligned} x[t] &= \varphi(x[t-1], v[t]), \quad x[0] = x_0, \\ y[t] &= \psi(x[t]), \quad t \in \mathrm{Z}_0 = \{0,1,2,\ldots\} \end{aligned} \tag{1.18}$$

где $v[t]$ – входы, $x[t]$ – состояния, $y[t]$ – выходы.

Общее окрестностное описание системы (1.18) имеет вид:

$$\begin{aligned} x[a] &= \Phi(x[O_x(a)], v[O_v(a)]), \\ y[a] &= \Psi(x[O_y(a)]). \end{aligned} \tag{1.19}$$

Симметричные окрестностные системы [9, 13, 17], например, удобно описывают, рассмотренные в [50] конструкции, вводимые в связи с методом конечных элементов. Другими примерами классических дискретных моделей, эквивалентных симметричным и смешанным окрестностным моделям, являются, представленные в [5, 6, 9, 13, 93, 107], метод сеток, сингулярная линейная модель, сингулярная линейная модель двумерной системы, клетки без памяти, клетки с памятью, одномерные однонаправленные и двунаправленные линейные итеративные цепи и другие.

Другим примером в рамках второго направления применения окрестностных систем к «вычислительным» математическим аппаратам и алгоритмам является представление сетей Петри (см. [42]) в виде недетерминированных динамических окрестностных систем, отраженные в работах [8, 12, 18, 75, 76, 81, 82, 87, 109]. Обобщенная сеть Петри задается определенной структурой и вектором начальной маркировки $C = (N, m_0)$:

1) $N = (P, T, F)$ – тройка, кодирующая структуру сети Петри C, где: $P = (p_1, p_2, \ldots, p_n)$ – все позиции сети Петри, $T = (t_1, t_2, \ldots, t_m)$ – все переходы

сети Петри (при этом автоматически $P \cap T = \emptyset$); F – набор дуг, соединяющих позиции и переходы и состоящий из дуг двух типов: $\{p_i, t_j\} \subseteq P \times T$ и $\{t_j, p_i\} \subseteq T \times P$;

2) $m_0 = (m_1^0, \dots, m_n^0)^{\mathrm{T}}$ – кортеж исходного состояния позиций сети Петри.

В представленных работах позициям сети Петри $P = (p_1, p_2, \dots, p_n)$ соответствуют узлы окрестностной системы $A = \{a_1, a_2, \dots, a_N\}$, маркировки позиций сети Петри – состояниям узлов окрестностной системы $X[a_i]$, начальная маркировка сети – состоянию окрестностной системы в начальный момент времени: $X[0] = m_0$. Потактовому изменению состояния сети Петри соответствует действие управлений $v[a_i,\ t]$ на узлы a_i (i = 1,…,n) окрестностной системы.

Набор дуг, соединяющих узлы A, состоит из m подмножеств $O[1]$, $O[2]$,…, $O[m]$, называемых слоями. Слою с номером k принадлежат узлы $A = \{a_1, a_2, ..., a_n\}$ (то есть каждый слой содержит все узлы) и те связи, которые определяются k-тым переходом. Например, $x[j] \in O[k]_{x[i]}$ и $v[j] \in O[k]_{v[i]}$, в том и только том случае, если $\{p_i, t_k\} \subseteq F$ и $\{t_k, p_j\} \subseteq F$ ($i = 1,..., n,\ j = 1,..., n$).

Для каждого слоя k авторами предлагается динамическая окрестносная модель сети Петри следующей структуры:

$$W_x^k[t+1]X[t+1] = W_x^k[t]X[t] + W_v^k[t]V[t] \qquad (1.20)$$

где $W_x^k[t+1], W_x^k[t], W_v^k[t]$ – три матрицы, содержащие коэффициенты соответствующего слоя k, первая – по состояниям для t+1, вторая – по состояниям для t и третья – и по входам для t.

Также, в рамках второго направления, в работах [37, 73, 116] авторы интерпретируют искусственные нейронные сети (ИНС) (см. [3, 19, 39, 66]) как окрестностные системы специальной структуры. А именно, предположим, что нейронная сеть имеет m слоев (в том числе входной и выходной слои), при этом j-тый слой ($j = 1,..., m$) имеет n_j нейронов. Тогда можно определить

окрестность $O_v[i^{(j)}]$ (по входам) нейрона $i^{(j)}$, то есть *i*-того нейрона, $i = 1,...,n_j$ в слое j в виде набора нейронов слоя $(j-1)$, имеющих связи с рассматриваемым нейроном (в случае полной сети прямого распространения это все нейроны предыдущего слоя). Заметим, что сами нейроны включаются в свои окрестности. Для нейронов входного слоя окрестности, очевидным образом, состоят из этих нейронов, $O_v[i^{(1)}] = \{i^{(1)}\}$, $i = 1,...,n_1$. Окрестность нейрона по состоянию совпадает с самим нейроном: $O_x[i^{(j)}] = \{i^{(j)}\}$.

Окрестностная модель ИНС имеет вид:

$$f\left(\sum_{\alpha \in O_v[a^j]} w_v[\alpha^{j-1}, a^j] v[\alpha^{j-1}]\right) = y[a^j], \qquad (1.21)$$

где $v[\alpha^j] \in R$ – вход в нейрон a слоя j ($j = 2,\ldots,m$) ИНС; $y[a^j] \in R$ – выход в нейроне a слоя j; $w_v[\alpha^{j-1}, a^j] \in R$ – матрица коэффициентов связей нейронов (j –1)-го слоя и нейрона a j-го слоя; $f : R \to R$ – функция активации.

Функция активации может быть линейной или нелинейной. В первом случае имеет место симметричная линейная окрестностная система.

В продолжение второго направления, в работах [11, 97, 98, 100, 115] были изложены примеры окрестностного подхода при моделировании разностных отображений, в частности, одномерного логистического отображения, двумерных отображений Ферхюльста-Пирла (подробнее – в [27]) и отображения Эно.

1.4. Точность аппроксимации вычислительных моделей и алгоритмов

Оценка прогностической точности тех или иных моделей, например, в машинном обучении – нейронных сетей и деревьев регрессии, рассматривалась во многих работах, например, в [19, 66, 43, 68, 127].

Производительность модели, ее прогностическая точность оценивается для того, чтобы узнать, насколько хорошо модель работает с новыми данными.

Алгоритм CART, ставший уже классическим в машинном обучении, был введен Лео Брейманом, Джеромом Фридманом и др. в работе [122]; разработки и модификации алгоритма проводились, например, Джоном Куинленом (см. [132, 133]) и другими учеными. Решающие деревья, подразделяющиеся на деревья классификации и деревья регрессии (алгоритм CART), представляют собой (чаще всего) бинарные деревья и содержат вершины двух типов – узлы (внутренние вершины дерева, включая корневой узел) и листья (концевые вершины). Связь узлов одного уровня иерархии дерева с узлами следующих уровней показана с помощью ориентированных дуг. Каждой внутренней вершине *v* дерева регрессии приписан определенный предикат (т.е. правило) $B_v(x, j, t): X \to 0,1,$ согласно которому текущее подмножество пространства признаков разбивается на две части, которые являются подмножествами пространства признаков следующих вершин дерева в структуре общей иерархии. Предикат $B_v(x, j, t)$ может иметь, в общем случае, произвольную структуру; в традиционном случае он может разбивать подмножество исходя из сравнения какого-либо *j*-го признака с пороговым значением $t \in \mathbb{R}$:

$$B_v(x, j, t) = [x_j \le t]. \qquad (1.22)$$

Каждой листовой вершине дерева *v* приписан прогноз $c_v \in Y$, где Y – область значений целевой переменной (таргета).

Для каждого объекта *x* движение в ходе предсказания, так же, как и при построении дерева, осуществляется сверху вниз, от корня к листьям. В очередной внутренней вершине *v* проход продолжится вправо, если $B_v(x) = 1$, и влево, если $B_v(x) = 0$. Проход продолжается до некоторого листа дерева, а ответом алгоритма в виде значения целевой переменной в зависимости от признака *x* будет прогнозное значение одной из листовой вершины c_v.

Дерево регрессии может быть легко переобучено: для этого достаточно построить такое дерево, в каждый лист которого будет попадать только один объект. Следовательно, при обучении деревьев регрессии, также, как и при обучении искусственных нейронных сетей, необходимо стремиться находить баланс между точностью и качеством прогноза.

Очевидно, что в вопросах предсказания непрерывных или категориальных значений, модель тем качественнее и эффективнее, чем точнее она предсказывает целевую переменную (таргет). В литературе разработано несколько способов и методов, позволяющих улучшить качество аппроксимации как при идентификации (обучении) модели, так и после завершения процесса обучения (см. [43, 68, 127]).

Основными направлениями улучшения базовых моделей CART являются ансамблиевые методы, такие как бэггинг (bagging), случайный лес (random forest), бустинг (boosting) и другие. В методе бэггинга используется усреднение k базовых алгоритмов деревьев регрессии $b_i(x), (i=1,...,k)$, обученных на k выборках:

$$a(x)=\frac{1}{k}(b_1(x)+...+b_k(x)). \qquad (1.23)$$

В методе бэггинга дисперсия композиции в k раз меньше дисперсии отдельного базового алгоритма. Однако, в данном методе не учитывается степень коррелированности базовых алгоритмов, при том, что они обучались на пересекающихся подвыборках одного и того же количества признаков. Идея случайного леса состоит в том, что в процессе обучения каждого дерева в каждой вершине случайно выбираются $n<N$ признаков, где N – полное число признаков (метод случайных подпространств). Это позволяет снизить корреляцию базовых алгоритмов друг на друга.

При бустинге деревьев регрессии алгоритмы учатся не параллельно, как при бэггинге, а последовательно: каждый следующий базовый алгоритм в бустинге обучается так, чтобы уменьшить общую ошибку всех своих предшественников. Таким образом, при бустинге каждый следующий

алгоритм (решающее дерево) в качестве таргета использует невязки (ошибки) предыдущего алгоритма. Общая ансамбливая модель имеет меньшее смещение, чем отдельное дерево регрессии, при этом дисперсия так же может уменьшаться.

Переобучением можно назвать ситуацию большой чувствительности модели к шумам и случайным выбросам, потерю моделью своих обобщающих свойств. Под недообученностью, наоборот, понимается малая чувствительность ко всему множеству данных, не только зашумленным, но и информативным и прогностически-полезным.

Переобучение модели становится основной проблемой для возможности прогнозирования на новых данных. Ошибка переобученной модели, рассчитанная для неизвестных данных, часто оказывается в разы больше ошибки на обучающем множестве. А значит эффективность прогнозирования модели, оцененная на обучающей выборке, не будет характеризовать эффективность (или качество прогноза) для новых неизвестных данных. Причины переобучения могут быть следующие:

1) мультиколлинеарность признаков;
2) наличие неинформативных признаков;
3) небольшое количество данных.

Справиться с мультиколлениарностью и, как следствие, переобученностью, помогают методы регуляризации, заключающиеся в добавлении дополнительных ограничений на условия, при которых возникают неадекватные значения весов в задачах регрессии, вызванные мультиколлинеарностью. Эти ограничения на вектор весов с некоторым параметром (коэффициентом) становятся частью функции потерь. Функцию потерь можно задать разными способами, наиболее известные – с помощью L^1-нормы и с помощью квадрата L^2-нормы вектора разницы исходных таргетов и предсказаний модели. В случае квадрата L^2-нормы функция потерь имеет вид:

$$L(f, X, y) = |y - f(X)|_2^2 = \|y - Xw\|_2^2 = \sum_{i=1}^{N} (y_i - \langle x_i, w \rangle)^2, \quad (1.24)$$

где X, y – объекты признаков и целевой переменной в выборке, N – объем выборки, w – веса модели регрессии $f(X) = \langle x_i, w \rangle$. В качестве функции потерь можно использовать и средний квадрат L^2-нормы, который традиционно называют среднеквадартическим отклонением (MSE).

$$L(f, X, y) = \text{MSE}(f, X, y) = \frac{1}{N} \sum_{i=1}^{N} (y_i - \langle x_i, w \rangle)^2. \quad (1.25)$$

Задача минимизации данного функционала в случаях с регуляризацией приводится к задаче минимизации функционала

$$\min_w L(f, X, y) = \min_w (\|y - Xw\|_2^2 + \lambda \|w\|_k^k), \quad (1.26)$$

где $\lambda \|w\|_k^k$ – регуляризационный член, λ – параметр (коэффициент) регуляризации, а k, как и в случае функции потерь, определяют или 1, или 2:

$$\begin{gathered} \|w\|_1^1 = |w_1| + |w_2| + ... + |w_D|, \\ \|w\|_2^2 = w_1^2 + w_2^2 + ... + w_D^2. \end{gathered} \quad (1.27)$$

В нейронных сетях техники регуляризации можно разделить на три обширные группы:

1) связанные с изменением функции потерь;
2) связанные с изменением структуры сети;
3) связанные с изменением данных.

Методы регуляризации решающих деревьев:

1) ограничение по максимальной глубине дерева;
2) ограничение на минимальное количество объектов в листе;
3) ограничение на максимальное количество листьев в дереве;
4) требование, чтобы функционал качества при делении текущей подвыборки на две улучшался не менее чем на s процентов.

Данные критерии можно проверять прямо во время построения дерева. Или можно построить дерево без ограничений и далее применить метод

стрижки (pruning), подразумевающий удаление некоторых вершины дерева. Суть данного метода в том, чтобы достичь условий регуляризации, при этом существенно не снижая качество итоговой модели. При этом для проверки качества модели необходимо (крайне желательно) иметь отложенную выборку, данные которой не использовались в процессе обучения исходного «неостриженного» дерева.

Одним из важнейших методов разработки адекватных моделей прогнозирования (как классификаторов, так и регрессионных), является метод кросс-валидации, или скользящего контроля, заключающийся в разбиении известного множества данных, как минимум на два подмножества. В случае разбиения на 2 части, одна из них используется для обучения модели, а другая для тестирования. Тестовая часть для уже обученной модели будет служить новым неизвестным набором данных. Сравнение для оценки точности происходит уже для целевой переменной из тестового множества. В другом случае, методе контроля по k-блокам, исходные данные делятся на подмножества тестовых. Данные случайным образом делятся на k непересекающихся подмножеств, обучение происходит на наборе из k-1 подмножеств, а тестовым является подмножество k. Таким способом циклически происходит обучение k различных моделей, каждый раз имея новое подмножество в качестве тестового. После этого предсказания этих моделей усредняются и сравниваются со значением целевой переменной для оценки точности.

Перечисленные способы и методы улучшения качества моделей и ее адекватности хорошо зарекомендовали себя на практике. Однако, в связи с разрабатываемыми в данной работе новым классом окрестностных моделей и новыми алгоритмами идентификации, имеющими определенные сходства с уже существующими деревьями регрессии, необходимо определять иные показатели качества и критерии остановки алгоритмов. Это связано с ситуацией, когда алгоритмы идентификации не предполагают методов кросс-валидации, т.е. разбиения исходного множества на обучающую и тестовую

части, а также в ситуациях, когда выборка для процесса идентификации мала. В рамках исследования для этих целей предлагается производить анализ остаточных данных вводимых моделей, подробно описанный в п. 3.4 третьей главы.

1.5. Постановка задач исследования

В предыдущих разделах данной главы представлены общие теоретические положения окрестностного моделирования и разработки конкретных классов окрестностных моделей. Показаны различные варианты применения окрестностных моделей, которые можно разбить на две большие группы – моделирование технологических/производственных процессов и применение как средство интерпретации и исследования некоторых математических методов и вычислительных алгоритмов. Описаны имеющиеся в литературе классы иерархических окрестностных моделей. Отмечается, что в одном случае иерархичность модели понималась как многослойность окрестностной структуры, где каждый отдельно рассматриваемый слой мог быть произвольной окрестностной моделью, а во втором – как введение дополнительных окрестностных моделей второго уровня внутри узлов первого уровня

Целью данного диссертационного исследования является разработка класса иерархических окрестностных моделей с древовидной окрестностной структурой, разработка методов и алгоритмов рекуррентной идентификации иерархических окрестностных моделей, методов анализа остаточных данных и применение разработанных алгоритмов в задачах моделирования процессов с нелинейным поведением.

В соответствии с целью исследования были поставлены и решены следующие задачи:

1. Обзор существующих классов окрестностных моделей и разработка моделей с иерархической окрестностной структурой.

2. Разработка алгоритмов и численных методов рекуррентной идентификации иерархических окрестностных моделей на основе использования остаточных данных выходов (невязок) промежуточных моделей.

3. Разработка метода контроля количества уровней в иерархических окрестностных моделях на основе анализа остатков промежуточных моделей.

4. Применение алгоритмов рекуррентной идентификации иерархических окрестностных моделей для исследования и прогнозирования свойств и характеристик процессов с нелинейным поведением.

5. Разработка проблемно-ориентированных модулей программного обеспечения, реализующих алгоритмы рекуррентной идентификации иерархических моделей.

2. РАЗРАБОТКА ИЕРАРХИЧЕСКИХ ОКРЕСТНОСТНЫХ МОДЕЛЕЙ

Данная глава посвящена разработке иерархических окрестностных моделей с древовидной окрестностной структурой. Такие структуры возникают в случае, если вершины соответствующего орграфа образуют дерево с двойными или одинарными дугами. Описаны способы кодирование вершин иерархических моделей, определены иерархические разбиения и иерархические кластеризации на множестве данных. Представлены иерархические окрестностные системы на древовидных структурах в общем случае и в случаях только восходящих или только нисходящих связей между узлами. Рассматривается возможность представления деревьев регрессии (в алгоритме CART) в виде иерархических окрестностных моделей. В связи с использованием иерархических моделей для интерпретации алгоритмов в главе обсуждаются два направления применения окрестностных моделей.

2.1 Два направления применения окрестностных моделей

Окрестностная модель или система состоит из окрестностной структуры и уравнений, связанных с узлами и выходами орграфа. Системы на графах, или системы уравнений, ассоциированные с графами или орграфами, часто появляются в различных приложениях и в различных вариантах, адаптированных к конкретным задачам (см. обзор в главе 1).

Как уже было указано во введении и в п. 1.3 первой главы, можно выделить два направления применения окрестностных моделей, технологическое и алгоритмическое.

Окрестностные системы являются универсальным способом формализации моделируемых объектов и производственных процессов. Окрестностной структурой, формализующей технологическую схему объекта, является орграф, оснащенный наборами данных для вершин или дуг; окрестностная модель – это система уравнений, в которой зависимости между

переменными задаются окрестностной структурой (см., например, [13, 15, 46-75]). Окрестностная структура, таким образом, в рамках первого направления, может соответствовать реальным узлам той или иной производственной цепочки, параллельных или последовательных операций, физически протекающих процессов с определенными наборами переменных и параметров. С другой стороны, в рамках второго направления, окрестностным системам и окрестностным структурам можно сопоставить определенные вычислительные методы и алгоритмы, такие как метод сеток, нейронные сети, сети Петри и т.д., т.е. алгоритмы, в которых имеется определенная дискретная структура, узлы и связи между ними. Во втором случае такие окрестностные системы могут не иметь под собой реального производственного воплощения, а рассматриваются как вычислительные алгоритмы на множествах данных произвольной природы.

Операторы вершин в структуре окрестностной модели могут быть полностью определены или известны только с точностью до настраиваемых параметров. Процесс нахождения параметров модели в технологическом варианте называется идентификацией; в алгоритмическом варианте вместо идентификации в некоторых случаях используется термин обучение.

В технологической ситуации обычно предполагается, что оператор, связанный с вершиной (узлом или выходом), линейно зависит от параметров. Кроме того, предполагается, что известен достаточно большой объем экспериментальных данных, а именно кортежей значений всех переменных («снимков» признаков и целевых переменных) в определенные моменты времени. Таким образом, для каждого узла (или выхода) существует набор данных входов и выходов, в связи с чем задача идентификации технологической окрестностной модели разбивается на независимые задачи регрессионной идентификации операторов отдельных узлов. Следует обратить внимание, что количество снимков (кортежей) должно быть не меньше максимального количества дуг, приходящих на один узел или выход, иначе задача идентификации будет не доопределена. В частности,

идентификация параметров вершинных операторов по одному снимку (кортежу), используемая в некоторых опубликованных работах, может рассматриваться лишь как минимальное по норме решение (среди бесконечного множества других) некорректной задачи.

В алгоритмической версии, т.е. когда алгоритм интерпретируется как окрестностная модель, процесс идентификации (обучения) зависит от конкретного алгоритма. Например, в методе конечных разностей вершинные операторы заранее известны из физических соображений или, более формально, вычисляются с помощью дискретизации непрерывных моделей. Для нейронных сетей прямого распространения вершинные операторы зависят от неизвестных параметров (весов) «почти» линейно, вся нелинейность сосредоточена в одномерных функциях активации. При наличии достаточного количества снимков модели можно было бы, как и в технологическом варианте, свести задачу к идентификации операторов отдельных узлов. Но суть нейронных сетей в том, что такие снимки невозможны, внутренние слои модели «спрятаны» и мы знаем только значения входных и выходных переменных. Задача идентификации решается с помощью достаточно сложного алгоритма обратного распространения ошибки (в сочетании с алгоритмом нелинейной оптимизации), который нетривиален даже в случае одного скрытого слоя.

После идентификации работа с окрестностной моделью в технологическом варианте обычно предполагает поддержание определенных номинальных режимов процесса с помощью синтеза регулятора для модели. В алгоритмическом варианте идентифицированная окрестностная модель фактически становится некоторой сложной формулой для вычисления выходов по входам.

2.2. Деревья, иерархические разбиения и иерархические кластеризации

Все вершины орграфа можно разделить на следующие классы. Вершины-входы не имеют предков, то есть не имеют входящих связей (дуг). Вершины-узлы имеют предков и потомков, то есть имеют входящие и выходящие связи. Вершины-выходы не имеют потомков, то есть не имеют выходящих связей. Эта классификация усложняется, если в орграфе разрешены петли, поскольку петли следует считать как входящими, так и выходящими связями для вершины с петлей. В данной работе орграфы с петлями не используются и потому приведенная классификация вершин достаточна. Орграфы могут иметь циклы, при этом можно различать ориентированные и неориентированные циклы.

В работе [45] разработано общее определение дерева как окрестностной структуры специального вида, описано кодирование вершин, определены иерархические разбиения и иерархические кластеризации. Дерево или, подробнее, ориентированное дерево – это орграф без петель и циклов, имеющий один вход (корень дерева) и один или более выходов, каждый из которых имеет только одну входящую дугу (листья дерева). Количество дуг, выходящих из вершины, не являющейся выходом, называется степенью разветвления (или ветвления) этой вершины. Число ориентированных дуг на пути от корня к данной вершине, увеличенное для удобства кодирования на единицу, называется уровнем этой вершины. Например, единственной вершиной уровня единица является корень дерева. Уровни выходов в общем случае могут быть разными. Если все выходы имеют один и тот же уровень k, то дерево называется k-уравновешенным, или k-деревом.

2.2.1. Кодирование вершин

Можно считать, что задана нумерация всех дуг, выходящих из каждой вершины дерева. А именно, для каждой вершины уровня r эти дуги

занумерованы числами $i_{r+1} = 1, 2, \ldots$. В этом случае любая вершина, начиная с вершин уровня $k > 2$ представляется последовательностью $(1, i_2, \ldots, i_k)$, в которой все числа (кроме первого, то есть кроме 1) соответствуют порядковым номерам связей на пути из корневой вершины в данную. Корневая вершина получает код (1). Тем самым определен порядок (лексикографический) всего множества вершин дерева. Выходы (листья дерева) наследуют этот порядок, при этом выходы с меньшим уровнем предшествуют выходам с большим уровнем. По кодам

$$\left\{ I(s) = \left(1, i_2(s), \ldots, i_{k(s)}(s)\right), \qquad s = 1, \ldots, S \right\} \tag{2.1}$$

выходов дерева T с $S = S(T)$ выходами, где $k(s)$ – это уровень выхода с номером s, дерево полностью восстанавливается.

Для $r \leq k(s)$ положим:

$$I_r(s) = \left(1, i_2(s), \ldots, i_r(s)\right). \tag{2.2}$$

Последовательность $I_r(s)$ является кодом r-той вершины, лежащей на пути, ведущем от корня к s-тому выходу. В частности, $I_{k(s)}(s) = I(s)$.

2.2.2. Иерархические разбиения

Отображение

$$P: D \to \{1, \ldots, S\} \tag{2.3}$$

произвольного множества D на множество выходов S (листьев) некоторого дерева T порождает систему подмножеств в множестве D, такую, что прообразы выходов $P^{-1}(s)$ последовательно объединяются по правилу, определяемому подъемом по ветвям дерева. Такая система подмножеств называется иерархическим разбиением множества D, порожденным отображением P.

Если в описанной выше конструкции множество D является конечным набором точек некоторого пространства M с заданной на нем метрикой, то

отображение (2.3) продолжается очевидным образом до отображения многогранников M_D (ячеек) Дирихле-Вороного (центрами которых являются точки множества D) , то есть

$$M_D = \{U(d) | d \in D\}. \quad (2.4)$$

В этой формуле подмножество $U(d)$ содержит все элементы из множества M, такие, что для них d – это самая ближняя точка из всех точек подмножества D; если таких точек несколько, то выбирается первая по порядку на D, такой порядок S предполагается заранее заданным. Следовательно, отображение (2.3) задает иерархическое разбиение на M_D и потому задает также иерархическое разбиение всего пространства M (это разбиение составлено из ячеек Дирихле-Вороного).

Введем следующее обозначение: $d(x)$ (где $x \in M$) – это номер многогранника Дирихле-Вороного, содержащего точку x. Далее, $s(x) = P(d(x))$ – это порядковый номер листа дерева T, в который отображается весь многогранник, содержащий x. Заметим, что в каждый выход (лист) дерева могут отображаться несколько многогранников и потому, в общем случае, может быть $P\big(d(x')\big) = P\big(d(x'')\big)$, даже если $d(x') \neq d(x'')$.

2.2.3. Иерархическая кластеризация

Элемент иерархического разбиения (любого уровня в иерархии) некоторого метрического пространства в общем случае может состоять из метрически далеких точек. Если каждый из элементов иерархического разбиения любого уровня состоит из метрически близких точек, то иерархическое разбиение называется иерархической кластеризацией. Данное определение, конечно, не является строгим, поскольку не фиксирована степень или свойство близости. Это связано также с тем, что понятие кластеризации (обычной, а не иерархической) допускает различные версии формализации (см. [29, 51]). Например, можно определить выпуклую

кластеризацию для подмножества $D \subset U \subset \mathbb{R}^n$, лежащего в некоторой области U эвклидова пространства $\mathbb{R}^n$. А именно, разбиение

$$D = D_1 \cup \ldots \cup D_r \tag{2.5}$$

является выпуклой кластеризацией множества D, если можно указать совокупность выпуклых подмножеств $\{C(D_i)\}$ в U, которые не пересекаются и каждое из которых содержит внутри себя только одно из подмножеств D_i. Это определение очевидным образом можно распространить на иерархический случай (выпуклое множество следующего уровня иерархии должно содержаться в выпуклом множестве предыдущего уровня). Простейшим примером выпуклой иерархической кластеризации является последовательность разбиений области U на подмножества, возникающее при вычислении интеграла Римана (как предела сумм Римана) по области U. Для любой (не обязательно выпуклой) кластеризации существует связанное с ней разбиение единицы, элементами которого являются нормализованные гауссианы, центры которых находятся в центрах эллипсоидов инерции кластеров. Вместо гауссианов можно использовать любые унимодальные плотности распределения (например, распределение Коши). Удобно сохранять термин «гауссиан» и в этом общем случае.

2.2.4. Иерархическое разбиение единицы

Многоуровневая схема. Для кластеров первого уровня разбиение единицы строится как обычно – берутся гауссианы для каждого кластера, затем каждый из них делится на сумму остальных, получаются нормализованные гауссианы. Для кластеров второго уровня разбиение единицы строится отдельно для каждого старшего кластера (первого уровня), то есть для каждого старшего кластера рассматриваются только младшие кластеры внутри данного старшего кластера, при этом построенное из них разбиение единицы дополнительно нормируется (умножается) на значения

уже нормализованного ранее гауссиана старшего кластера. Далее процесс распространяется вниз по всей иерархии.

Одноуровневая схема. Можно рассматривать более простую одноуровневую схему, когда разбиение единицы для кластеров очередного уровня строится для всех сразу, без учета принадлежности старшим кластерам иерархии. Это означает, что при нормализации гауссиана очередного уровня он делится на сумму всех гауссианов всех кластеров данного уровня, а не на сумму гауссианов внутри старшего кластера, как в многоуровневой схеме.

2.3. Иерархические окрестностные структуры и модели

Наиболее общее определение следующее: окрестностная структура называется иерархической, если вершины соответствующего орграфа образуют дерево с двойными или одинарными дугами, восходящими и нисходящими и, кроме того, для корня и для каждой терминальной вершины могут иметься как вход, так и выход. Пример такой структуры показан на рисунке 2.1.

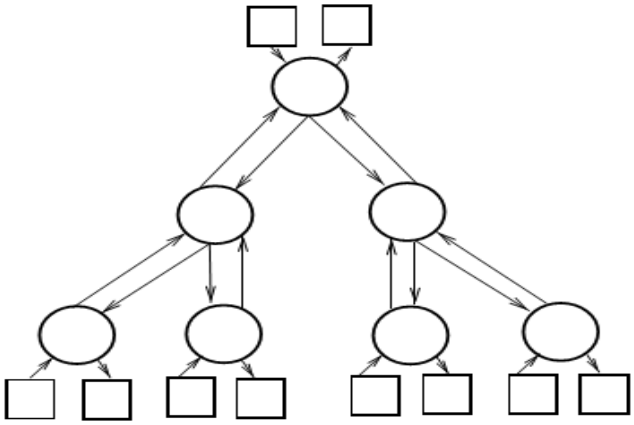


Рисунок 2.1 – Пример иерархической окрестностной структуры

Для целей дальнейших исследований будет удобно немного изменить определение окрестностной модели. Будем считать, что оснащение орграфа состоит из переменных, связанных с дугами (а не с вершинами, как в предыдущем случае) и, кроме того, с каждым внутренним узлом (с

вершинами, не являющимися входными или выходными) связан оператор, преобразующий набор входящих переменных в множество исходящих переменных. Если для каждого узла все переменные, связанные с исходящими дугами, одинаковы, новое определение окрестностной модели совпадает с предыдущим. Окрестностную структуру такой модели будем называть иерархической.

Для описания иерархической окрестностной модели, соответствующей иерархической окрестностной структуре, будем использовать следующие обозначения. Входные и выходные данные корневого узла — это $u(0)$ и $y(0)$. Корневой узел — $x(i_1) = x(1)$. Далее, рекуррентно, $x(i_1, \dots, i_s, i_{s+1})$ является i_{s+1}-м узлом из упорядоченного набора узлов, подчиненных узлу $x(i_1, \dots, i_s)$. Входными и выходными данными терминальных узлов $x(i_1, \dots, i_s, i_{s+1})$ являются $u(i_1, \dots, i_s, 1)$ и $y(i_1, \dots, i_s, 1)$. Количество узлов, подчиненных узлу $x\ (i_1, i_2, \dots, i_s)$, обозначается $n(i_1, i_2, \dots, i_s)$. В частности, на рисунке 2.1 имеем $n(1) = n(1,1) = n(1,2) = 2$ и $n(1,1,1) = n(1,1,2) = n(1,2,1) = n(1,2,2) = 0$. Обозначения переменных аналогичны обозначениям узлов. Каждая переменная, относящаяся к дуге, направленной вверх (соответственно вниз), обозначается так же, как начальная (соответственно конечная) вершина соответствующей дуги, с заменой буквы на заглавную и индексами u (соответственно d). Далее оператор, соответствующий узлу $x(i_1, \dots, i_s)$, удобно представить в виде двух операторов F и G для дуг, направленных вверх и вниз. Пусть $I_s = (i_1, \dots, i_s)$. Уравнения иерархических окрестностных моделей с k уровнями в явном динамическом варианте имеют вид:

$$\begin{cases} Y_u(0) = X_u^{t+1}(1) = F_1\left(X_u^t(1,1), \dots, X_u^t\big(1, n(1)\big)\right); \\ [X_d^{t+1}(1,1), \dots, X_d^{t+1}(1, n(1))] = \\ G_1\left(X_u^t(1,1), \dots, X_u^t\big(1, n(1)\big), U_d^t(0) = X_d^t(1)\right); \\ X_u^{t+1}(I_s) = F_{I_s}\left(X_u^t(I_s, 1), \dots, X_u^t\big(I_s, n(I_s)\big), X_d^t(I_s)\right); \\ [X_d^{t+1}(I_s, 1), \dots, X_d^{t+1}(I_s, n(I_s))] = \\ G_{I_s}\big(X_u^t(I_s, 1), \dots, X_u^t(I_s, n), X_d^t(I_s)\big); \\ s = 2, \dots, k-1\ ; \\ X_u^{t+1}(I_k) = F_{I_k}\big(U_u(I_k, 1), X_d^t(I_k)\big); \\ Y_d^{t+1}(I_k, 1) = G_{I_k}\big(X_d^t(I_k)\big). \end{cases} \tag{2.6}$$

Если иерархическая модель содержит только дуги вниз (соответственно только дуги вверх), система уравнений будет иметь вид

$$\begin{cases} [X_d^{t+1}(1,1), \dots, X_d^{t+1}(1, n(1))] = G_1\big(X_d^t(1)\big); \\ [X_d^{t+1}(I_s, 1), \dots, X_d^{t+1}(I_s, n(I_s))] = G_{I_s}\big(X_d^t(I_s)\big); \\ s = 2, \dots, k-1\ ; \\ Y_d^{t+1}(I_k, 1) = G_{I_k}\big(X_d^t(I_k)\big) \end{cases} \tag{2.7}$$

и, соответственно,

$$\begin{cases} Y_u(0) = X_u^{t+1}(1) = F_1\left(X_u^t(1,1), \dots, X_u^t\big(1, n(1)\big)\right); \\ X_u^{t+1}(I_s) = F_{I_s}\left(X_u^t(I_s, 1), \dots, X_u^t\big(I_s, n(I_s)\big)\right); \\ s = 2, \dots, k-1\ ; \\ X_u^{t+1}(I_k) = F_{I_k}\big(U_u(I_k, 1)\big), \end{cases} \tag{2.8}$$

Примеры иерархических окрестностных структур, соответствующих иерархической модели с дугами вниз (u-модель) и иерархической модели с дугами вверх (d-модель), показаны, соответственно, на рисунке 2.2 и рисунке 2.3.

Если все переменные одномерны, то в «d-модели» все уравнения имеют вид $G\colon \mathbb{R}^1 \to \mathbb{R}^n$, а в «$u$-модели» все уравнения имеют вид $F\colon \mathbb{R}^n \to \mathbb{R}^1$.

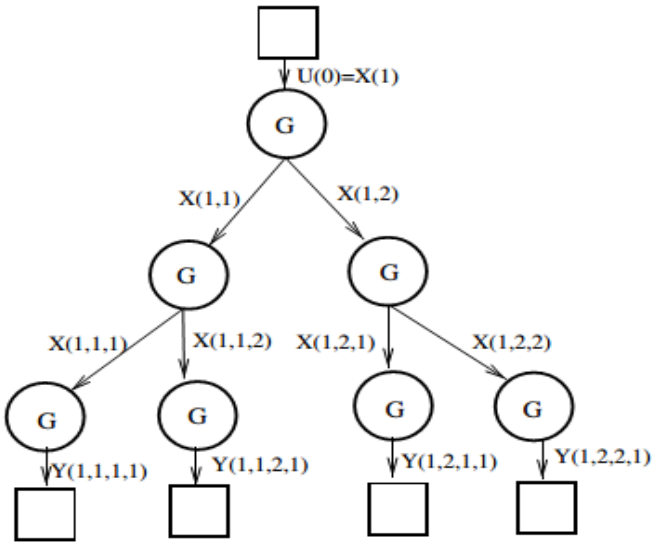


Рисунок 2.2 – Пример иерархической d-модели

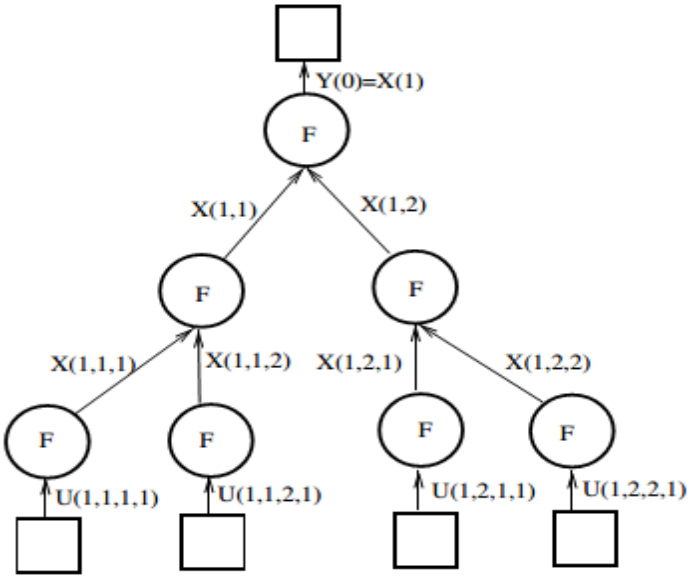


Рисунок 2.3 – Пример иерархической u-модели

Если все переменные $X_d(I_s,1),\dots,X_d(I_s,n(I_s))$, относящиеся к нисходящим дугам от одного узла, совпадают и поэтому могут быть обозначены проще как $X_d(I_s)$. В частности, в иерархических организационных системах переменную $X_d(I_s)$ можно интерпретировать как общий для всех подчиненных узлов «приказ», записанный на основе набора полученных «отчетов» $X_u^t(I_s,1),\dots,X_u^t\big(I_s,n(I_s)\big)$ снизу и «приказа» $X_d(I_{s-1}(I_s)))$ сверху. Таким образом, операторы F генерируют «отчеты», операторы G генерируют «заказы». Модель будет иметь вид:

$$\begin{cases} Y_u(0) = X_u^{t+1}(1) = F_1\left(X_u^t(1,1), \dots, X_u^t\big(1, n(1)\big)\right); \\ X_d^{t+1}(1) = G_1\left(X_u^t(1,1), \dots, X_u^t\big(1, n(1)\big), U_d^t(0)\right); \\ X_u^{t+1}(I_s) = F_{I_s}\left(X_u^t(I_s, 1), \dots, X_u^t\big(I_s, n(I_s)\big), X_d^t(I_{s-1}(I_s))\right); \\ X_d^{t+1}(I_s) = G_{I_s}\big(X_u^t(I_s, 1), \dots, X_u^t(I_s, n), X_d^t(I_{s-1}(I_s))\big); \\ s = 2, \dots, k-1\,; \\ X_u^{t+1}(I_k) = F_{I_k}\big(U_u(I_k, 1), X_d^t(I_{k-1}(I_k))\big); \\ Y_d^{t+1}(I_k, 1) = G_{I_k}(X_d^t(I_{k-1}(I_k)))\,. \end{cases} \tag{2.9}$$

где $I_{s-1}(I_s) = (i_1, \dots i_{s-1})$ для данного $I_s = (i_1, \dots, i_s)$.

2.4. Деревья регрессии как иерархические окрестностные модели

Решающие деревья, подразделяющиеся на деревья классификации и деревья регрессии (алгоритм CART), представляют собой бинарные иерархические деревья и содержат вершины двух типов – узлы (внутренние вершины дерева, включая корневой узел) и листья (концевые вершины) (подробнее – в главе 1).

Были выявлены значительные сходства алгоритма CART (рассматривается только алгоритм деревьев регрессии как часть CART) с разработанными иерархическими окрестностными моделями, а именно: дерево регрессии как алгоритм идентификации регрессионной модели является частным случаем более общей иерархической окрестностной модели. Движение по дереву регрессии осуществляется «сверху-вниз», т.е. от корня к листьям, как при построении дерева на обучающих и тестовых выборках, так и при прогнозировании на новых выборках, то есть связь между узлами однонаправленна. Узлы дерева содержат предикат, или решающее правило, в соответствии с которым происходит разделение узла-предка на два узла-потомка, т.е. правило разбиения множества значений тех или иных признаков, попавших в узел. Такое содержательное значение узлов не предполагает наличия петель.

Следовательно, структура дерева регрессии представляет собой связный ориентированный граф без петель и без циклов, и в общем случае может быть представлена в виде иерархической окрестностной структуры, как показано в параграфе 2.2, а сама модель дерева регрессии – в виде иерархической оркестностной модели. Дадим описание структуры такой модели с помощью кодирования вершин и движения по дереву (см. параграф 2.1).

Пусть дерево регрессии состоит из n уровней, на каждом уровне r $(r=1,...,n)$ находится m_r узлов. При $r=1$ имеем корневой узел, а для всех $r \geq 2$ значение m_r четно. Степень ветвления не концевой вершины в таком дереве всегда равна 2. Две выходящие дуги каждой вершины уровня r можно пронумеровать числами $i_{r+1}=1,2$, и тогда каждая вершина уровня $k \geq 2$ кодируется последовательностью чисел $(1,i_2,...,i_k)$. В случае k-уравновешенного дерева каждый лист кодируется последовательностью одной и той же длины. Корень дерева является вершиной уровня 1 и имеет код (1). Например, в случае 3-уравновешенного дерева, т.е. когда корень имеет два узла-потомка, каждый из которых в свою очередь имеет два листа-потомка (см. рисунок 2.4), кодировка каждой из $m_3=4$ концевых вершин представляют собой последовательности $(1,1,1)$, $(1,1,2)$, $(1,2,1)$, $(1,2,2)$.

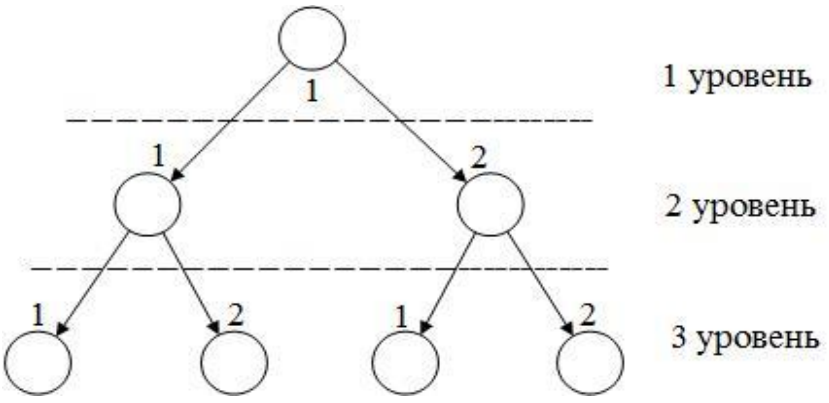


Рисунок 2.4 – Пример 3-уравновешенного дерева регрессии в виде окрестностной структуры с кодировкой дуг $i_{r+1}=1,2$

При этом дерево регрессии не должно обязательно быть *k*-уравновешенным, алгоритмы CART, так же, как и иерархическое окрестностные модели, допускают маршруты движения по дереву разной длины. Весь набор кодов выходов (листьев) полностью определяет движение по дереву регрессии в задачах предсказания. Построенное дерево, в соответствии с предикатами в каждом узле, разбивает множество значений обучающей выборки $\{x_i\}_{i=1}^{N} \in \mathbb{R}^{N\times M}$ (где M – количество исходных признаков, N – объем выборки) на подмножества, представляющие собой области пространства $U \subset \mathbb{R}^M$, ограниченные гиперплоскостями. В случае, например, пороговых предикатов такие области будут представлять собой прямоугольники в $\mathbb{R}^2$, прямоугольные параллелепипеды в $\mathbb{R}^3$ и т.д.

Если обозначить за $i^{(r)}$ i-ый узел r-го уровня ($i^{(r)} = 1,...,m_r$), то окрестностью такого узла по входу $O_v[i^{(r)}]$ будет узел-предок предыдущего $(r-1)$-го уровня. Окрестность узла по состоянию совпадает с самим узлом: $O_x[i^{(r)}] = \{i^{(r)}\}$.

Таким способом можно представить дерево регрессии в виде иерархической окрестностной модели, в частности, иерархической d-модели, система уравнений которой представлена по формуле (2.7).

2.5. Выводы по главе 2

В главе 2 получены следующие результаты:

1. Описан способ кодирования вершин деревьев моделей, иерархические разбиения в виде сюрьективного отображения некоторого множества D на конечное множество выходов $\{1,...,S\}$ дерева T, и иерархические кластеризации в виде иерархических разбиений множества D, каким-либо образом учитывающее близость элементов (точек) D по метрике.

2. Описаны две схемы разбиения единицы – многоуровневая и одноуровневая. При многоуровневой схеме построенное разбиение единицы младших кластеров внутри старшего кластера дополнительно нормируется (умножается) на значения уже нормализованного ранее гауссиана старшего кластера. В одноуровневой схеме при нормализации гауссиана очередного уровня он делится на сумму гауссианов всех кластеров данного уровня, а не на сумму гауссианов внутри старшего кластера.

3. Разработаны иерархические окрестностные структуры, определенные в случаях, когда вершины соответствующего орграфа образуют дерево с двойными или одинарными дугами, восходящими и нисходящими, при этом для корня и для каждой терминальной вершины могут иметься как вход, так и выход.

4. Разработаны иерархические окрестностные модели с соответствующей иерархической окрестностной структурой. При этом оператор любого узла такой модели представим в виде двух операторов F и G для дуг, направленных вверх и вниз. Если модель имеет дуги, направленные только вниз, то такую модель называем «d-моделью» (операторами узлов становятся только операторы G). Если модель имеет дуги, направленные только вверх, то такую модель называем «u-моделью» (операторами узлов становятся только операторы F).

5. Описано представление алгоритмов деревьев регрессии как части алгоритма CART в виде иерархических окрестностных моделей.

3. РАЗРАБОТКА АЛГОРИТМОВ РЕКУРРЕНТНОЙ ИЕРАРХИЧЕСКОЙ ИДЕНТИФИКАЦИИ

В данной главе представлены разработанные алгоритмы рекуррентной иерархической идентификации и схема анализа остаточных данных промежуточных моделей в алгоритмах рекуррентной идентификации, позволяющая контролировать количество уровней иерархии, необходимое для достижения соответствия между точностью и качеством прогноза иерархической модели. Первый алгоритм предполагает наличие заранее заданной иерархической кластеризации или, в более общем случае, разбиения множества входных данных (объекта в целом или его отдельных узлов). Второй алгоритм реализует схему синтеза модели, связанную с кластеризацией множества невязок, получаемых в процессе последовательного уточнения исходной модели. В этом случае, вместо заранее заданной иерархической кластеризации или разбиения, на множестве входных данных возникает ассоциированное с кластеризацией невязок иерархическое разбиение.

Для первого алгоритма описывается схема идентификации общей непрерывной модели с помощью иерархического разбиения единицы. Второй алгоритм дополнен численным методом, основанным на трихотомии невязок промежуточных моделей.

Описывается схема агрегирования локальных и микролокальных мод окрестностной системы в мультимодальную окрестностную систему.

3.1. Статический алгоритм (R-алгоритм) рекуррентной иерархической идентификации

В работах [45, 111] вводится схема рекуррентной идентификации окрестностных моделей на основе некоторой априорной (заранее заданной) иерархической кластеризации множества кортежей входных данных. Данную

схему предлагается рассматривать как алгоритм рекуррентной иерархической идентификации окрестностных моделей, названный R-алгоритмом, так как имеется отдаленное сходство с классическими конструкциями интеграла Римана. Статическим алгоритм можно назвать в связи с тем, что кодируемая деревом T, иерархическая кластеризация множества входных данных (независимых переменных) задается заранее, в отличие от алгоритма, предполагающего построение дерева динамически вместе с процессом идентификации.

3.1.1. Идентификация общей кусочно-непрерывной модели

R-алгоритм рекуррентной иерархической идентификации:

1. Задается некоторая иерархическая кластеризация кортежей входных данных $D \subset \mathbb{R}^n$, кодируемая деревом T, и ассоциированная с ней иерархическая кластеризация области определения $U \subset \mathbb{R}^n$ входов модели. Каждый из элементов иерархической кластеризации области U состоит из многогранников Дирихле-Вороного.

2. Идентифицируется некоторая начальная модель $F^1(x)$ по всем данным $D \subset \mathbb{R}^n$.

3. Вычисляются остатки (невязки) этой начальной модели:

$$\hat{d}^1 = \hat{d} - F^1(d), \tag{3.1}$$

где $\hat{d} \in \mathbb{R}^1$ – это значение выхода для входа $d \in \mathbb{R}^n$.

4. Идентифицируются уточняющие модели $F^2_{I_2}(x)$ следующего 2-го уровня, выходами для которых служат невязки первого уровня $\hat{d}^1$, ассоциированные с соответствующими входами $d \in \mathbb{R}^n$ для каждого элемента кластеризованного разбиения второго уровня $D^2_1, D^2_2, ..., D^2_{r_2}$, $I_2 = 1, ..., r_2$ – код вершины дерева второго уровня.

5. Процесс итерируется: для каждой модели элементов разбиения второго уровня $D_1^2, D_2^2, ..., D_{r_2}^2$ вычисляются остатки моделей второго уровня и идентифицируются дополнительные модели $F_{I_3}^3(x)$ для элементов разбиения третьего уровня иерархии $D_1^3, D_2^3, ..., D_{r_3}^3$ и так далее для следующих уровней кластеров иерархии.

Итоговая иерархическая окрестностная модель:

$$F(x) = F^1(x) + F_{I_2}^2(x) + F_{I_3}^3(x) + \cdots + F_{I_k}^k(x), \qquad (3.2)$$

где $k = k(s(x))$ и $I_2 = I_2(s(x)), I_3 = I_3(s(x)), \ldots, I_k = I(s(x))$ – коды вершин уровней 1, 2,…, *k*, лежащих на пути от концевой вершины к корню.

Свойства модели (3.2):

(a) эта модель является кусочно-непрерывной;

(b) компактная формула (3.2) на самом деле скрывает внутри себя довольно сложную конструкцию, поскольку для вычисления значения $F(x)$ требуется сначала вычислить номер выхода $s(x)$. В общем случае для этого может потребоваться перебор по всем кортежам входов $D \subset \mathbb{R}^n$; формулу (3.2) можно назвать «иерархическим рядом», что отличает ее от обычных суммируемых слагаемых в рядах тем, что слагаемые в виде уточняющих моделей $F_{I_i}^i(x)$ изменяются для уровней иерархии i = 2,…, k, в зависимости от движения по дереву модели T;

(c) корректирующие слагаемые в формуле (3.2) могут иметь различную природу, так как для общей иерархической модели (3.2) важным является наличие остатков у каждого из корректирующих слагаемых, а не их конкретная природа и вид. В частности, эти слагаемые могут являться линейными или полиномиальными регрессиями, регрессионными моделями других видов (нелинейными по объясняемым параметрам, нелинейные по факторам – синусоидальные, экспоненциальные и др. зависимости), моделями деревьев регрессий, моделями нейронных сетей и т.д. В случае, если слагаемые являются классическими линейными регрессиями, то общую

модель (3.2) можно назвать иерархической квазилинейной регрессионной моделью. При этом, если корректирующие слагаемые идентифицировать, например, как однородные полиномы второй степени для второго уровня, третьей степени для третьего уровня и так далее, то в такой версии модель может рассматриваться как нелокальный иерархический аналог формулы Тейлора;

(d) при достижении желаемой точности на очередном кластере иерархии, в случае чрезмерной «глубины» изначально построенного дерева T для данной ветки дерева, можно удалить все последующие вершины из дерева, при этом формула (3.2) для модели $F(x)$ сохраняется, поскольку в этой формуле не требуется, чтобы дерево T было уравновешенным. И наоборот, в случае недостаточной «глубины» дерева для какой-либо ветки, можно идентифицировать общую модель, остановить алгоритм, произвести необходимую дополнительную иерархическую кластеризацию и, возобновив алгоритм в соответствии с пунктом 5, идентифицировать новую, дополненную новыми корректирующими слагаемыми, модель (3.2).

Согласно свойству (b), при использовании модели с целью прогнозирования, необходимо вычислять номер выхода $s(x)$, т.е. определять необходимый лист дерева T, соответствующий нужному кластеру многогранников Дирихле-Вороного. Если точка входного кортежа, подлежащая прогнозированию, из кластеризованной области определения $U \subset \mathbb{R}^n$ расположена внутри одного из этих многогранников, то для данного входа модель (3.2) вычисляет значение выхода в соответствии с той веткой, которая заканчивается данным выходом $s(x)$.

R-алгоритм удобен в ситуациях, когда обучающее множество входных данных имеет ярко выраженную (например, визуально на диаграмме рассеяния) исходную кластеризацию.

Схема работы R-алгоритма представлена на рисунке 3.1.

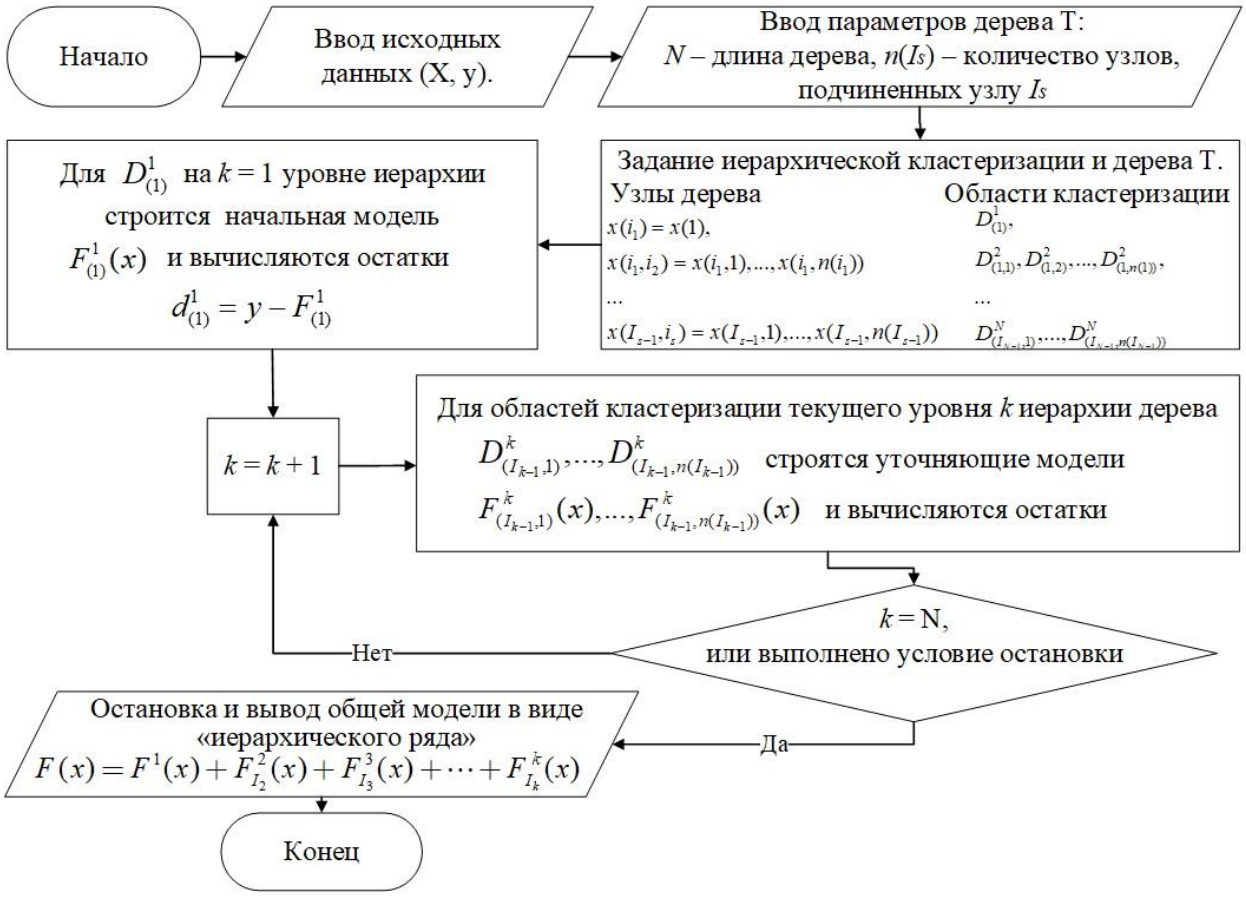


Рисунок 3.1 – Схема работы R-алгоритма рекуррентной идентификации

После идентификации применение модели для прогнозирования, во-первых, происходит «снизу-вверх», т.е. от листьев к корню, в отличие от многих известных регрессионных алгоритмов, таких как, например, нейронные сети и алгоритмы CART (в части деревьев регрессии), а во-вторых, необходим поиск ближайшей точки среди обучающих кортежей (что также отличает применение модели от нейросетей и CART). Нахождение ближайшей точки означает попадание в многогранник Дирихле-Вороного данной точки и, соответственно, попадание в тот или иной лист дерева T, выбор конкретной регрессионной модели среди множества моделей, скрывающегося за структурой общей кусочно-непрерывной иерархической модели (3.2). Нахождение ближайшей точки в задаче прогнозирования осуществляется с помощью классической евклидовой метрики.

3.1.2. Идентификация общей непрерывной модели

Как было показано в [45], иерархическая кластеризация является выпуклой, поэтому слагаемые в кусочно-непрерывной модели (3.2) с помощью гауссианов иерархического разбиения единицы (одноуровневого или многоуровневого) можно объединить в непрерывную модель $\tilde{F}(x)$. Таким образом, для построения непрерывной модели $\tilde{F}(x)$ требуется предварительное вычисление гауссианов. При этом под термином «гауссиан» здесь понимается плотность любого унимодального многомерного распределения, например, Гаусса или Коши.

Далее, при использовании модели $\tilde{F}(x)$, не требуется находить номер выхода $s(x)$, и потому объем вычислений по формуле $\tilde{F}(x)$ существенно меньше, чем в случае кусочно-непрерывной модели $F(x)$.

В качестве примера приведем формулы для $\tilde{F}(x)$ в случае, когда T – двоичное дерево с тремя уровнями. Для одноуровневой схемы модель имеет вид:

$$\begin{gathered}\tilde{F}(x)=F^1(x)+\varphi_{11}^2(x)F_{11}^2(x)+\varphi_{12}^2(x)F_{12}^2(x)+\\ +\varphi_{111}^3(x)F_{111}^3(x)+\varphi_{112}^3(x)F_{112}^3(x)+\varphi_{121}^3(x)F_{121}^3(x)+\varphi_{122}^3(x)F_{122}^3(x),\end{gathered} \tag{3.3}$$

для многоуровневой схемы вид модели следующий:

$$\begin{gathered}\tilde{F}(x)=F^1(x)+\varphi_{11}^2(x)[F_{11}^2(x)+\varphi_{111}^3(x)F_{111}^3(x)+\varphi_{112}^3(x)F_{112}^3(x)]+\\ +\varphi_{12}^2(x)[F_{12}^2(x)+\varphi_{121}^3(x)F_{121}^3(x)+\varphi_{122}^3(x)F_{122}^3(x)].\end{gathered} \tag{3.4}$$

Нормализованные гауссианы φ_{1i}^2 второго уровня с одинаковыми нижними индексами в формулах (3.3) и (3.4) совпадают. Нормализованные гауссианы φ_{1ij}^3 с одинаковыми нижними индексами в формулах (3.3) и (3.4) происходят из одинаковых гауссианов, но нормализующие множители у них разные. А именно,

$$\varphi_{1ij}^3=\frac{\bar{\varphi}_{1ij}^3(x)}{\bar{\varphi}_{111}^3(x)+\bar{\varphi}_{112}^3(x)+\bar{\varphi}_{121}^3(x)+\bar{\varphi}_{122}^3(x)} \tag{3.5}$$

в формуле (3.3) и

$$\varphi_{1ij}^{3} = \frac{\bar{\varphi}_{1ij}^{3}(x)}{\bar{\varphi}_{1i1}^{3}(x) + \bar{\varphi}_{1i2}^{3}(x)} \tag{3.6}$$

в формуле (3.4). Здесь $\bar{\varphi}_{1ij}^{3}(x)$ – это исходные ненормализованные гауссианы соответствующих кластеров.

Непрерывная версия R-модели, агрегированная с применением нормализованных «гауссианов» по формулам (3.3) или (3.4), может быть интерпретирована как иерархическое обобщение модели Такаги-Сугено-Канга (см. [131, 134]).

3.2. Динамический алгоритм (L-алгоритм) рекуррентной иерархической идентификации

В работах [45, 112, 117, 118] приводятся разработки L-алгоритма иерархической идентификации, который в некотором смысле имитирует конструкцию интеграла Лебега. После построения исходной (начальной) модели рассматривается кластеризация множества невязок (то есть точек на прямой). Для каждого кластера находятся прообразы невязок в пространстве входов и идентифицируется уточняющая модель второго уровня. Далее процесс итерируется, то есть для каждой из уточняющих моделей последнего уровня (из уже построенных) рассматривается кластеризация невязок этой модели, прообразы в пространстве входов для каждого кластера и идентифицируются уточняющие модели следующего уровня.

В этой конструкции последовательные кластеризации в пространстве значений модели (то есть на прямой) соответствуют некоторому дереву T. Заметим, что множества кластеров (на прямой) при этом не образуют иерархии, то есть не образует иерархического разбиения в определенном в п. 2.1 смысле, поскольку на каждом шаге множество невязок изменяется и кластеризации происходят каждый раз на разных множествах точек на прямой. С другой стороны, в пространстве входных данных $D \subset \mathbb{R}^n$, очевидно,

возникает иерархическое разбиение, образованное прообразами кластеризованных невязок на прямой и кодируемое деревом *T*. Взаимное расположение элементов этого разбиения в $D \subset \mathbb{R}^n$, в общем случае, может быть сколь угодно сложным (именно в этом проявляется аналогия с интегралом Лебега).

3.2.1. Идентификация общей кусочно-непрерывной модели

L-алгоритм рекуррентной иерархической идентификации следующий:

1. Идентифицируется некоторая начальная модель $F^1(x)$ по всем данным $D \subset \mathbb{R}^n$.

2. Вычисляются остатки (невязки) этой начальной модели:

$$\hat{d}^1 = \hat{d} - F^1(d). \qquad (3.7)$$

3. Проводится кластеризация остатков $\hat{d}^1$ модели. Получаем ассоциированное с кластеризованными значениями остатков иерархическое разбиение пространства входов $D = D_1^2 \cup D_2^2 \cup ... \cup D_{r_2}^2$.

4. Идентифицируются уточняющие модели второго уровня $F_{i(x)}^2(x)$ – модели каждого иерархического разбиения входов, для которых выходной величиной служат соответствующие кластеризованные остатки.

5. Процесс итерируется: для каждой из дополнительных уточняющих моделей последнего уровня *k* (из уже построенных) проводится кластеризация невязок этой модели, находятся прообразы в пространстве входов для каждого кластера и идентифицируются уточняющие модели следующего уровня.

Схема работы L-алгоритма представлена на рисунке 3.2.

Формула (3.2) для итоговой модели вместе с расшифровкой обозначений $k = k(s(x))$ и $I_2 = I_2(s(x)), I_3 = I_3(s(x)), \ldots, I_k = I(s(x))$ остается формально такой же, как в R-алгоритме, поскольку эта формула также является «иерархическим рядом» для L-алгоритма, т.е. уточняющие слагаемые

определяются движением по дереву *T*. Также сохраняются и свойства (a)-(d), определенные ранее для R-алгоритма.

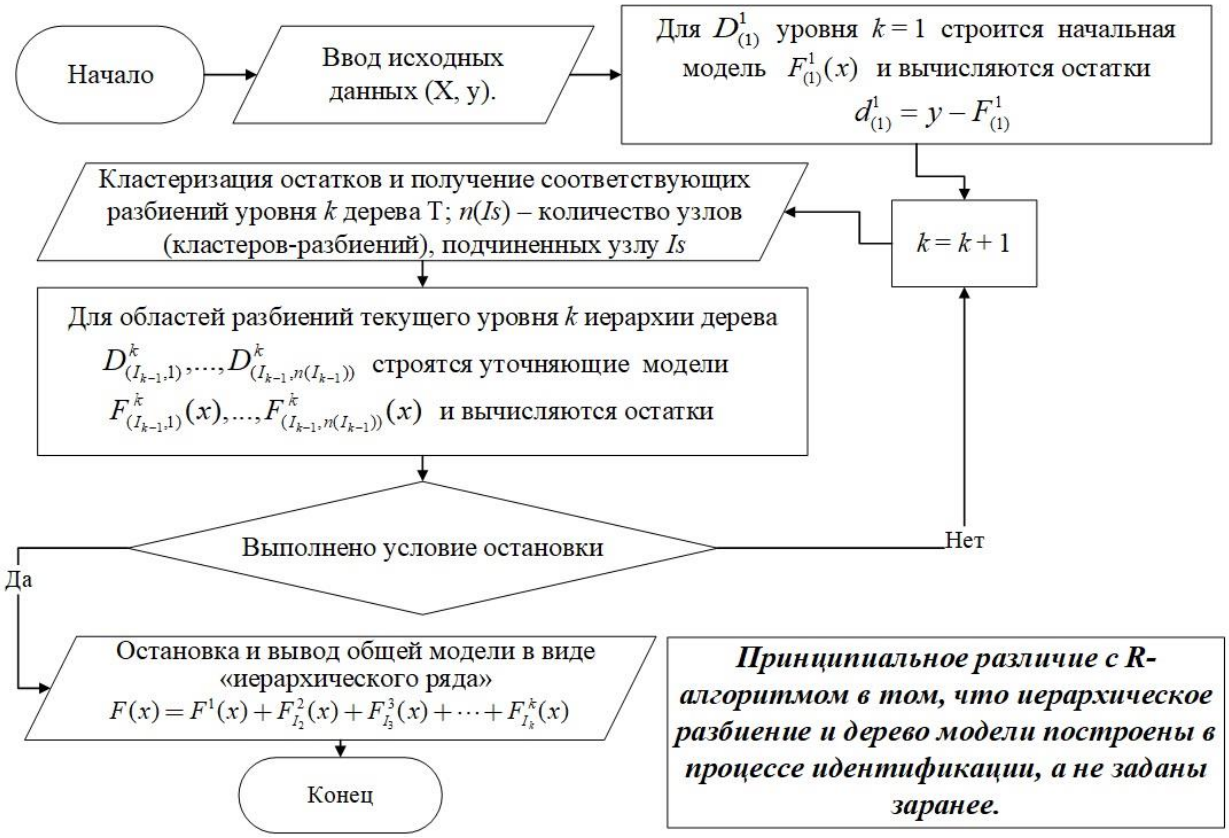


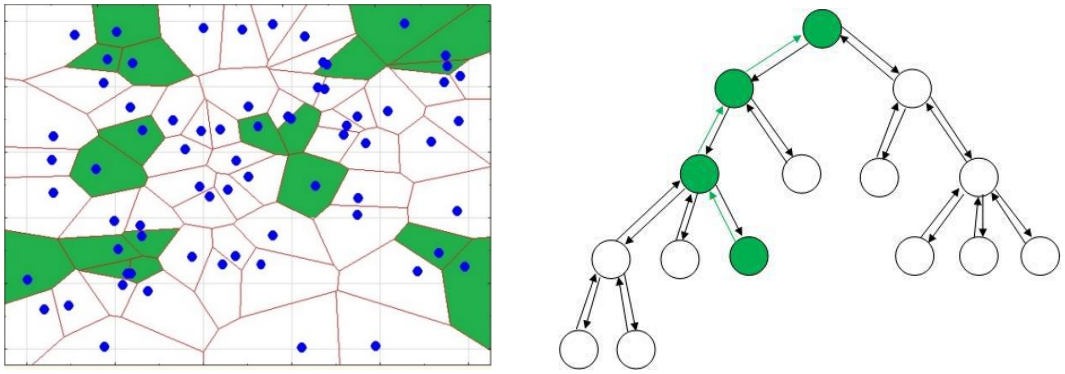
Рисунок 3.2 – Схема работы L-алгоритма рекуррентной идентификации

Отличительной особенностью L-алгоритма, как от R-алгоритма, так и от многих алгоритмов предсказаний непрерывной величины, является то, что дерево модели T строится вместе с процессом идентификации каждой дополнительной уточняющей модели $F^2_{I_2}(x), F^3_{I_3}(x),\ldots,F^k_{I_k}(x)$. В связи с этим алгоритм называется динамическим. Процесс идентификации общей иерархической модели останавливается вместе с построением дерева *T*, и наоборот.

Еще одним отличием от многих других алгоритмов, в том числе и от R-алгоритма, является то, что иерархическое разбиение пространства входов $D \subset \mathbb{R}^n$, ассоциированное с кластеризованными невязками на прямой, в общем случае, имеет сколь угодно сложную структуру, т.е. данные разбиения входных данных не являются связными и выпуклыми множествами.

Элементами этих множеств при этом также являются ячейки Дирихле-Вороного.

На рисунке 3.3 показаны многогранники (ячейки) Дирихле-Вороного (а), дерево иерархической модели (б). Использование модели для прогнозирования, как и в случае R-алгоритма, происходит «снизу вверх», т.е. от листьев к корню, при этом необходим поиск ближайшей точки среди обучающих кортежей.



а) б)

Рисунок 3.3 – Иерархическое разбиение множества входных данных (а) и соответствующее ему движение по дереву «снизу-вверх» в задаче прогнозирования по модели (б), обозначенные зеленым цветом

Зеленым цветом выделена одна из концевых вершин дерева и путь к корню, а также ячейки, ассоциированные с зеленой концевой вершиной (отображенные в нее). В одной из таких ячеек находится точка нового кортежа, подлежащая прогнозированию.

3.2.2. L-алгоритм с трихотомией невязок

Описанная конструкция L-алгоритма предполагает существование кластеров невязок на каждом шаге аппроксимации. Если явно выраженных кластеров нет, то уточнение модели в L-алгоритме можно осуществить, например, следующим образом. Выбирается некоторое число ε (точность аппроксимации) и после построения линейной модели разбивается множество

входных данных $D \subset \mathbb{R}^n$ по величине соответствующих невязок на три подмножества по формуле:

$$\begin{aligned} D &= D_\varepsilon \cup D^- \cup D^+, \\ D_\varepsilon &= \{ d \in D;\ \left| \hat{d} - F^1(d) \right| < \varepsilon \}, \\ D^- &= \{ d \in D;\ \hat{d} < F^1(d) - \varepsilon \}, \\ D^+ &= \{ d \in D;\ \hat{d} > F^1(d) + \varepsilon \}. \end{aligned} \tag{3.8}$$

Для множества D^ε процесс построения модели закончен (требуемая точность уже достигнута), для множеств D^- и D^+ идентифицируются уточняющие модели. Далее процесс повторяется. Каждая не концевая вершина дерева T, согласно данному алгоритму, имеет три выходящие дуги, одна из которых ведет в концевую вершину, которой отвечает достижение требуемой точности модели на соответствующем элементе разбиения. В данном алгоритме дерево T не является уравновешенным (за исключением тривиального двухуровневого случая – листья дерева выходят сразу из корня).

Визуально иерархическое разбиение на основе трихотомии остатков показано на рисунке 3.4, где слева показана синим, красным и зеленым цветом трихотомия остатков на прямой и, справа, – ассоциированное с этой трихотомией разбиение пространства входных данных.

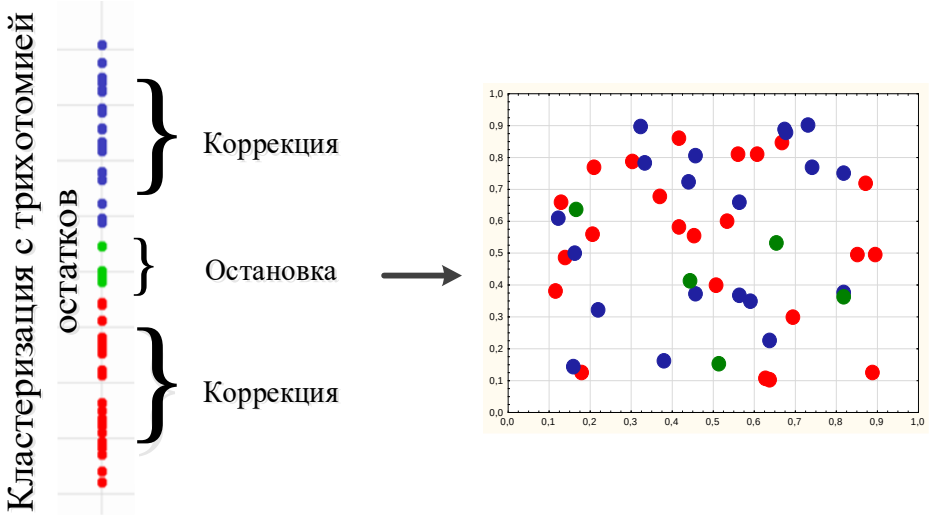


Рисунок 3.4 – L-алгоритм с трихотомией невязок

Для синего и красного множеств производится коррекция – проводится дальнейшая идентификация уточняющих моделей согласно L-алгоритму.

Как уже отмечалось в параграфе 2.4, алгоритмы (CA)RT (деревья регрессии) можно интерпретировать как иерархические окрестностные модели, точно так же, как и L-алгоритм. При этом, помимо некоторых сходств, есть и определенные различия в этих алгоритмах. Сходство состоит в том, что деревья моделей строятся посредством процесса идентификации от корня до листьев, т.е. их можно интерпретировать как иерархическую d-модель. В каждом узле генерируется формула, согласно которой данные разбиваются на группы и передаются вниз по дереву.

Различие же в процессе идентификации следующее: в (CA)RT дерево является двоичным, а при идентификации узлов используются только входные данные; выходы используются только в листьях как аргументы некоторой функции, например, среднего значения. В L-алгоритме с трихотомией, например, дерево является троичным (а в общем случае, корень и не концевой узел могут содержать и более трех потомков), а при идентификации корня и каждого узла используются как входные, так и выходные данные.

В режиме вычисления по моделям сходство заключается только в наличии дерева в окрестностной структуре. Вычисление в (CA)RT идет по схеме d-модели сверху вниз, как и идентификация. В L-алгоритме вычисление идет по схеме u-модели, то есть снизу вверх.

Также отмечаются сходства и различия L-алгоритма с бустингом деревьев регрессии. В методе бустинга, как и в L-алгоритме, модель уточняется с использованием остатков, при этом слагаемые модели находятся с помощью алгоритма (CA)RT, т.е. каждое слагаемое итоговой модели в бустинге является деревом (при этом, как правило, довольно простым, например двоичным пнем). В итоге в бустинге модель задается обычным рядом, каждое слагаемое которого «иерархично» и соответствует дереву. В отличие от L-алгоритма, у которого иерархичен сам ряд.

3.3. Численные методы анализа исходных и остаточных данных в задачах иерархической идентификации

3.3.1. Анализ исходных данных

Как отмечалось в работе [88] на примере производства клинкера и в работе [114], многие технологические процессы и их параметры можно рассматривать во временной составляющей, обусловленной естественными измерениями в разные последовательные моменты времени. С этой точки зрения значения этих параметров можно рассматривать как временные ряды. Исследованию временных рядов различных процессов (нелинейных во времени) посвящено множество работ, так как данная теория является эффективным инструментом для анализа таких процессов в различных областях наук (см., например, [2, 26-28, 30, 40, 41, 44, 52-53, 67, 69-71]).

С целью обнаружения непериодических, хаотических явлений необходимо использовать определенные методы анализа временных рядов, т.е. необходимы некоторые числовые показатели (критерии), которые можно использовать для сравнения экспериментальных и теоретических значений, определить степень хаотичности и детерминированности данных.

В 1971 году Рюэль и Такенс (см. [26]) ввели математический образ динамического хаоса – странный аттрактор. Для характеристики аттракторов введено понятие размерности (см. [41]). Известно, что размерность странного аттрактора является дробной (фрактальной). Определить размерность аттрактора можно следующим образом: вводят некоторое количество n-мерных кубиков (ячеек) $N(\varepsilon)$ с ребром ε. Тогда $N(\varepsilon)$ – минимальное число n-мерных ячеек, необходимых для покрытия этого множества, зависящее от ε следующим соотношением (см. [71]):

$$N(\varepsilon) \cong \frac{1}{\varepsilon^{D}}, \tag{3.14}$$

где D – размерность рассматриваемого множества.

Чем меньше ε, тем точнее будет значение размерности D:

$$D = \lim_{\varepsilon \to o} \frac{\ln N(\varepsilon)}{\ln(1/\varepsilon)} . \qquad (3.15)$$

Фрактальная размерность не учитывает, какое количество раз точки данного множества посещали ту или иную ячейку. Такая мера, которая бы взвешивала каждый непустой элемент покрытия с помощью относительной частоты, с которой он посещается типичной траекторией, относятся к вероятностным, а размерности называются вероятностными. К их числу относят, например, корреляционную размерность D_c (см. [71]):

$$D_c = \lim_{\varepsilon \to 0} \frac{\ln C(\varepsilon)}{\ln \varepsilon} = \frac{\ln\left(\sum_{i=1}^{N(\varepsilon)} p_i \right)}{\ln \varepsilon} , \qquad (3.16)$$

где p_i – вероятность того, что пара точек аттрактора принадлежит i-ой ячейке.

Для вычисления корреляционной размерности и реконструкции аттрактора на практике используют алгоритм Грассбергера-Прокаччиа (см. [71]). Имеется выборка N значений наблюдаемой дискретной переменной x_i ($i \in 1 \ldots N$). Для реконструкции аттрактора необходимо задать некоторые целые значения p и m ($p = 1,2,3,\ldots$; $m = 1,2,3,\ldots$). Тогда можно построить m-мерный вектор, компонентами которого являются значения дискретной переменной в моменты времени i, $i+p$, $i+2p$, $i+(m-1)p$ (см. [71]):

$$\mathbf{z}_i = \left\{ x_i, x_{i+p}, \ldots, x_{i+(m-1)p} \right\}. \qquad (3.17)$$

В данном случае величина p называется временной задержкой, а m – размерностью вложения. Для заданного p и каждой размерности вложения m рассчитывают оценку значения $C(\varepsilon)$ по формуле (см. [71]):

$$C(\varepsilon) = \sum_{i=1}^{N(\varepsilon)} p_i = \lim_{N \to \infty} \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j}^{N} \theta\left(\varepsilon - \left\| \mathbf{z}_i - \mathbf{z}_j \right\| \right) \equiv C_m(\varepsilon) , \qquad (3.18)$$

где $i, j \in 1 \ldots N$; θ – функция Хевисайда.

Величина $C_m(\varepsilon)$ называется корреляционным интегралом, она является оценкой значения $C(\varepsilon)$ при достаточно больших значениях выборки N. Как видно из формулы (3.16) значение D_c находится из следующего соотношения:

$$\ln C_m(\varepsilon) = D_c \ln \varepsilon \,, \tag{3.19}$$

в котором вместо $C(\varepsilon)$ используется ее оценка $C_m(\varepsilon)$.

На практике, ввиду особенностей рассматриваемых систем и аттракторов, при малых и больших значениях ε могут в логарифмических координатах получаться нелинейные участки. Поэтому строят зависимость $\ln C_m(\varepsilon)$ от $\ln\varepsilon$ в широком диапазоне по ε и «на глаз» ищут на ней наиболее линейный участок, наклон которого и будет определять значение корреляционной размерности D_c. Значение D_c вычисляют для каждой размерности вложения m. В полностью детерминированных системах при увеличении m размерность D_c будет сходиться к своей истинной величине, а та размерность вложения, начиная с которой D_c перестает меняться, является наименьшей целой размерностью пространства, содержащего весь аттрактор.

Несколько иначе обстоит дело в системах с наличием белого шума. В таких системах корреляционная размерность увеличивается с ростом размерности вложения.

В [119] приводятся результаты расчетов по оценке влияния на рост корреляционной размерности мультипликативной случайной компоненты ряда с дисперсией, равной дисперсии исходного ряда. Для этого рассматриваются ряды известных нелинейных уравнений, порождающих детерминированный хаос. В числе этих уравнений – генератор Ван дер Поля, отображение Эно, отображение Икеды, система Лоренца и др. Для каждого из перечисленных детерминированных рядов G_k (элементы которых обозначим g_i, $i = 1,\ldots,N$) создаются два ряда, содержащие в разной пропорции случайную компоненту:

- полностью хаотизированный случайный ряд (полученный путем случайного перемешивания исходного ряда). Обозначим его S_k с элементами s_i, $i = 1,\ldots,N$;

• частично детерминированный ряд $G^{\alpha}S^{\beta}$, имеющий α% детерминированного хаоса и β% случайного. Элементы ряда получаются по следующей формуле (см. [119]):

$$z_i = g_i^{\alpha} \cdot s_i^{\beta}, \quad i = 1,\ldots,N, \quad \alpha + \beta = 1. \tag{3.20}$$

Для всех рядов находится последовательность корреляционных размерностей, соответствующих размерностям вложений. Далее для каждого ряда по этим корреляционным размерностям строится линейная регрессия. В результате была установлена зависимость между коэффициентами наклона полученных регрессий и долей β% случайного хаоса в частично детерминированных рядах. Данная зависимость выражается уравнением (см. [119]):

$$Y = -1{,}13X^2 + 1{,}98X, \tag{3.21}$$

где независимой переменной X является значение коэффициента наклона прямой регрессии последовательности корреляционных размерностей D_c при различных пространствах вложения m, а зависимой переменной Y – доля β% случайного хаоса. Оказывается, что просматривается тенденция к увеличению наклона в регрессии корреляционных размерностей при возрастании случайной хаотической компоненты в ряду наблюдений (см. [119]).

Одним из методов анализа нелинейной динамики является метод нормированного размаха (*R/S*-анализ). Впервые данный метод предложил в своей работе [124] Х.Е. Херст. Алгоритм вычисления постоянной Херста.

Для ряда наблюдений $E = \{e_i\}$ (i = 1,…,N) выбирается промежуток времени n и организуется цикл по k от 1 до (N-n+1). При k = 1 выбираем первые n наблюдений e_1, e_2,…, e_n и вычисляем их среднее значение M_{1n}.

Строится временной ряд накопленных отклонений:

$$X_{1m} = \sum_{i=1}^{m} (e_i - M_{1n}), \quad m < n. \tag{3.22}$$

При k = 2 выбираем следующие n наблюдений e_2, e_3,…, e_{n+1} и вычисляем их среднее значение M_{2n} и ряд X_{2m}:

$$X_{2m} = \sum_{i=2}^{m+1} (e_i - M_{2n}) . \quad (3.23)$$

Повторяя процесс для всего цикла по k получаем (N-n+1) значений кумулятивных отклонений X_{km}, для которых вычисляем размах по формуле:

$$R_n = \max(X_{km}) - \min(X_{km}) . \quad (3.24)$$

Херст ввел следующее соотношение:

$$R/S = (an)^H , \quad (3.25)$$

где S – среднеквадратическое отклонение рассматриваемого ряда; a –константа из интервала (0, 1); H – постоянная Херста.

Прологарифмировав (3.25) получим:

$$\ln(R/S) = H \ln(an) . \quad (3.26)$$

Откладывая по оси y значения $\ln(R/S)$, а по оси x – $\ln(an)$ получим точки, по которым строим линейную регрессию. Наклон полученной линии регрессии дает значение постоянной Херста H.

Значения и смысл постоянной Херста следующие:

- если $H \cong 0{,}5$, то временной ряд состоит из последовательности случайных независимых (некоррелированных) величин. Это так называемый «белый шум» с максимальной хаотичностью;
- если $0 \le H < 0{,}5$, то это свидетельство антиперсистентности (неустойчивого состояния) ряда, характеризуемого так называемым «розовым шумом». Т.е. если такой ряд возрастал в предыдущий период, то он скорее всего будет убывать в последующий период, и наоборот;
- если $0{,}5 > H \ge 1$, то ряд соответствует «черному шуму», т.е. ряд имеет долгосрочную память, он будет персистентным, а при значениях H заметно превосходящих 0,5 еще и трендоустойчивым.

Для рассмотренных в данном пункте таких показателей хаотичности, как оценка корреляционной размерности и доля случайного хаоса, была

разработана программа для ЭВМ (см. [105]), применяемая, например, в работе [112].

3.3.2. Анализ остаточных данных

При анализе остаточных данных предлагается, рассмотренный в работах [112, 118], метод использования автокорреляционной функции (АКФ) остатков иерархических моделей для остановки построения дерева модели *T*. Данный метод основывается на следующей идее: как известно из литературы (см., например, [49, 64]), АКФ какого-либо ряда (сигнала) показывает корреляцию между значениями сигнала и значениями копии этого сигнала, сдвинутого на определенный шаг (лаг), от величины шага. Традиционно данные сигналы рассматриваются как некоторые функции времени или случайные процессы, и, соответственно, сдвиг для расчета АКФ происходит на определенное значение времени $\tau \geq 1$. Сигналы, состоящие из сложных колебаний, могут быть проанализированы с помощью АКФ, позволяя увидеть периодичность в ряде данных. Отсутствие же каких-либо автокорреляций для разных значений $\tau \geq 1$ или их малая значимость может быть свидетельством распределения такого ряда, как близкого к «белому шуму».

При идентификации иерархических окрестностных моделей с помощью R-алгоритма и L-алгоритма необходимо выделить существенную составляющую моделируемого процесса таким образом, чтобы, с одной стороны, ошибка аппроксимации была как можно меньше, а с другой стороны, избежав при этом переобучения, иначе модель потеряет обобщающие способности. Для этого предлагается после идентификации уточняющих моделей на каждом новом уровне иерархии, т.е. при последовательном увеличении глубины дерева модели, проводить анализ остатков общей иерархической модели с помощью АКФ. При этом предлагается остатки выстраивать в порядке, соответствующем возрастанию (или убыванию) выходной величины (таргета).

Очевидно, что до начала моделирования и идентификации, сами значения выходной величины, расположенные в порядке возрастания (или убывания) имеют АКФ близкую к 1 для всех лагов τ. Значения же АКФ невязок иерархической модели будут отличаться от 1. С увеличением числа уровней в иерархической модели, ошибка аппроксимации будет снижаться, модель будет точнее приближать значения целевой переменной, соответственно невязки будут уменьшаться по абсолютной величине, при этом их распределение, упорядоченное в соответствии с возрастанием таргетов, будет терять «внутреннюю информацию». Об этом процессе будет свидетельствовать АКФ невязок. При достижении АКФ невязок, которая мало отличима от АКФ для «белого шума», алгоритм (R или L) предлагается остановить, требуемая точность общей иерархической модели достигнута.

Близость распределения невязок к «белому шуму» в качестве оценки адекватности модели и остановки алгоритмов идентификации можно считать обоснованным исходя из следующих соображений: при таком подходе модель отражает существенную информацию, заложенную в данных, игнорируя несущественные, хаотически распределенные шумы. Иными словам, анализ остаточных данных в задачах идентификации иерархических окрестностных моделей можно рассматривать в том числе как валидацию моделей.

Анализ остаточных данных на наличие «белого шума» можно проводить в том числе методами, описанными в предыдущем пункте 3.3.1.

3.4. Идентификация мультимодальных окрестностных систем

В задачах построения окрестностных моделей сложных распределенных производственных объектов нередко встречается ситуация, когда имеется несколько технологически разрешенных (номинальных) режимов функционирования объекта, зависящих, например, от кластеров входных данных. В этом случае соответствующая окрестностная система должна быть мультимодальной, то есть должна описывать все номинальные режимы

производственного объекта как «локальные моды» окрестностной системы с возможностью переходов между ними (см. работы [113, 126]). Каждый номинальный режим объекта является набором номинальных режимов его технологических узлов. Соответственно, каждой локальной моде мультимодальной системы отвечает некоторый набор «микролокальных» мод узлов окрестностной структуры, моделирующей распределенный объекта. Предлагается две схемы построения мультимодальной окрестностной системы: на основе идентификации локальных мод системы в целом (с последующим агрегированием этих мод в общую систему) и на основе агрегирования микролокальных мод узлов, без идентификации локальных мод всей системы. Вторую схему можно использовать для синтеза мультимодальной окрестностной системы из уже имеющихся мультимодальных подсистем для узлов.

3.4.1. Окрестностные системы на вершинах и дугах орграфа

Для дальнейшего будет удобно дополнительно потребовать, чтобы каждый выход окрестностной структуры, то есть вершина без выходящих дуг, имел только одну входящую дугу. В общем случае для этого достаточно формально добавить к окрестностной структуре некоторое количество новых вершин-выходов, чтобы превратить в узлы (с одной выходящей дугой) все выходы с двумя и более входящими дугами. Окрестностной системой называется система уравнений, ассоциированная с окрестностной структурой, при этом правила написания таких уравнений могут варьироваться в зависимости от степени детализации технологической схемы. Максимальной детализации (которая не всегда удобна) соответствуют вертексные окрестностные системы или системы на вершинах орграфа (см. [46, 48]), когда переменные соответствуют вершинам, а уравнения соответствуют узлам орграфа, при этом все выходящие дуги данного узла передают одну и ту же переменную, скалярную или векторную. Наиболее близкими к реальным

технологическим схемам обычно являются реляционные системы (см. [48, 106]), или системы на дугах орграфа, когда переменные соответствуют дугам, а уравнения соответствуют выходящим дугам узлов орграфа, при этом разные выходящие дуги узла или входа передают, вообще говоря, разные переменные. С помощью преобразования окрестностной структуры (декластеризации, см. [47]) всегда можно перейти от реляционной системы к вертексной. С другой стороны, любая вертексная система формально является также и реляционной.

В задаче идентификации мультимодальных окрестностных систем рассматриваются окрестностные системы на дугах (то есть реляционные системы), поскольку они более удобны при агрегировании нескольких подсистем в общую систему. Тип системы (статическая или динамическая) и конкретный вид ее уравнений при этом, как правило, не имеет значения, и потому будет удобно использовать короткую абстрактную запись реляционной окрестностной системы в виде уравнения

$$\bar{X}_{-} = \bar{\mathbf{F}} \circ \bar{X}_{+}, \tag{3.27}$$

где $\bar{X}_{+}$ – метавектор входов в узлы X (вектор метаисточников), $\bar{X}_{-}$ – метавектор выходов из узлов X (вектор метастоков) и $\bar{\mathbf{F}}$ – вектор операторов преобразования входящих переменных в выходящие (размерности метавекторов $\bar{X}_{-}$, $\bar{X}_{+}$, и вектора $\bar{\mathbf{F}}$ равны количеству узлов окрестностной структуры). В частности, каждая компонента метавектора $\bar{X}_{+}$ соответствует некоторому узлу и является кортежем всех (в общем случае векторных) переменных, входящих в этот узел по дугам. Аналогично, каждая компонента метавектора $\bar{X}_{-}$ соответствует некоторому узлу и является кортежем всех (в общем случае векторных) переменных, выходящих из этого узла по дугам. Размерность каждой компоненты метавектора $\bar{X}_{+}$ равна количеству дуг, входящих в соответствующий узел, размерность каждой компоненты метавектора $\bar{X}_{-}$ равна количеству выходящих дуг. Для вертексных систем

(систем на вершинах) размерность каждой компоненты метавектора $\bar{X}_{-}$ равна единице, то есть это одна (в общем случае векторная) переменная. Произведение $\circ$ (операторное произведение Адамара) – это покомпонентное действие вектора операторов $\bar{\mathrm{F}}$ на метавектор метаисточников $\bar{X}_{+}$, результатом действия является метавектор $\bar{X}_{-}$. В случае динамических систем нужно еще добавить верхние индексы t+1 и t переменным в левой и правой частях уравнений; для упрощения записи в данном случае их можно не добавлять.

Сформулированное выше условие для выходов (добавление формальных вершин-выходов) позволяет не записывать отдельно уравнения для выходов Y, как это обычно делается в записи систем управления, поскольку эти уравнения уже содержатся в системе $\bar{X}_{-} = \bar{\mathrm{F}} \circ \bar{X}_{+}$ как уравнения для соответствующих дуг. При этом, для описания вертексных и реляционных окрестностных систем удобен язык метаграфов (см. [121]). Такое описание было дано, например, в [47], но в данной работе метаграфы не используются.

3.4.2. Локальные моды окрестностной системы в целом

Номинальными режимами будем называть режимы распределенного объекта, заданные технологическими инструкциями. Модальной идентификацией окрестностной системы называем описание номинальных режимов моделируемого объекта вместе с последующей идентификацией, как правило, регрессионной, соответствующих этим режимам локальных мод окрестностной системы. Локальность в данном случае понимается как локальность по данным входов, то есть каждая локальная модель или локальная мода описывает поведение моделируемого распределенного объекта в определенном диапазоне (кластере) значений входов. Достаточно часто технологическая схема предусматривает несколько номинальных

режимов и возможность переключения между ними при изменении входных параметров или текущей задачи производства. Синтез нелинейной динамической модели, например, полиномиальной, описывающей процесс со всеми локальными и переходными режимами, как правило, является слишком сложной задачей. В то же время для описания поведения объекта вблизи номинального режима, то есть для локальной моды, обычно вполне достаточно линейной модели. Поэтому имеет смысл рассмотреть схему агрегирования нескольких линейных моделей в некоторую «квазилинейную» окрестностную систему. Близкая по смыслу задача решается, например, в рамках систем нечеткого управления Такаги-Сугено (см. [131, 134]). Далее описывается версия подобной схемы агрегирования в контексте окрестностных систем.

3.4.3. Микролокальные моды окрестностной системы

В предыдущем пункте рассматривались локальные модели (локальные моды) окрестностных систем и задача описания всех номинальных (то есть технологически допустимых) режимов вместе с переходами между ними. Предполагается, что номинальные режимы объекта так или иначе известны заранее и задача моделирования состоит в построении локальных моделей для соответствующих кластеров входных данных и дальнейшем агрегировании этих локальных моделей. Заметим, что локальные модели, несмотря на название, описывают объект в целом, то есть функционирование всех его узлов, локализуется только диапазон данных. В задаче синтеза окрестностной системы из уже построенных подсистем, которые соответствуют узлам окрестностной структуры, эти подсистемы могут, в свою очередь, иметь свои «микролокальные» моды, соответствующие номинальным режимам подсистемы.

Может оказаться, что нам заранее известны только микролокальные моды подсистем (узлов), в то время как локальные моды объекта в целом

неизвестны. В этом случае будет удобна математическая модель, основанная на микромодальной идентификации и последующем агрегировании микролокальных моделей в общую модель. В частности, это позволяет исследовать вопрос о том, какие локальные режимы сложного объекта могут возникать при совместном функционировании составляющих его подсистем. При этом в общем случае не всякий набор микролокальных мод будет порождать локальную моду системы в целом. Теоретически возможны ситуации, когда, несмотря на наличие микролокальных мод, система в целом не будет иметь хорошо выраженных кластеров в пространстве входных переменных, которым соответствовали бы локальные моды. Кроме того, возможны случаи возникновения локальных мод, соответствующих технологически бесполезным или даже нежелательным локальным режимам функционирования распределенного объекта.

3.4.4. Агрегирования локальных мод

После идентификации локальных моделей нужно объединить полученные локальные окрестностные системы в общую мультимодальную систему с параметрами, зависящими от значений входов и состояний. В этой схеме не имеет значения способ получения локальных систем, регрессионный или основанный на каких-либо физических законах, но требуется информация о разбиении (кластеризации) множества значений входов и переменных состояний на области, соответствующие локальным системам. Опишем дискретную и непрерывную версии мультимодального объединения локальных мод.

Пространство входов разбивается на кластеры, и актуальная для данного набора входных данных локальная мода задается соответствующим кластером. В дискретной версии мультимодальная система представляет собой линейную комбинацию локальных моделей, при этом коэффициентами являются характеристические функции некоторого разбиения пространства

входных данных, соответствующего кластерам. Переходы между локальными модами в этом случае будут происходить скачкообразно. В непрерывной схеме вместо характеристических функций используются непрерывные (на самом деле – гладкие) функции, образующие разбиение единицы, связанное с кластерами входных данных. Переходы между локальными модами в непрерывном случае происходят без скачков.

Входы U можно разделить на внешние неуправляемые $\hat{U}$ и внутренние управляемые $\tilde{U}$. Исходные экспериментальные данные (датасет) $D = D^U, D^X, D^Y = \hat{D}, \tilde{D}, D^X, D^Y$ – это кортежи состояний системы, то есть наблюдаемые значения неуправляемых и управляемых входов, внутренних переменных и выходов. Если размерность пространства внешних входов равна $\hat{N}$, то $\hat{D} \subset \mathbb{R}^{\hat{N}}$. Заданным номинальным режимам соответствует кластеризация множества входных данных $\hat{D}$, то есть $\hat{D} = \hat{D}_1 \cup ... \cup \hat{D}_S$. Пусть $d_1, ..., d_S \subset \mathbb{R}^{\hat{N}}$ – центры кластеров. Локальные модели, соответствующие номинальным режимам, запишем в виде $\bar{X}_- = \bar{\mathrm{F}}^r \circ \bar{X}_+$, где $r = 1, ..., S$. Агрегированной окрестностной системой называется система, заданная формулой

$$\bar{X}_- = \alpha_1(d)\bar{\mathrm{F}}^1 \circ \bar{X}_+ + ... + \alpha_S(d)\bar{\mathrm{F}}^S \circ \bar{X}_+, \qquad (3.28)$$

где $d \subset \mathbb{R}^{\hat{N}}$ и $\alpha_1(d) + ... + \alpha_S(d) = 1$, $\alpha_r \geq 0$ ($r = \overline{1,s}$). Выбор функций $\alpha_r(d)$ в дискретном и непрерывном агрегировании может быть сделан следующим образом.

В дискретном случае удобно использовать разбиение Дирихле-Вороного (математические пакеты, как правило, содержат модули построения такого разбиения). Пусть $\mathbb{R}^{\hat{N}} = \mathfrak{D}_1 \cup ... \cup \mathfrak{D}_S$ – разбиение Дирихле-Вороного пространства $\mathbb{R}^{\hat{N}}$, порожденное центрами кластеров $\hat{D} = \hat{D}_1 \cup ... \cup \hat{D}_S$. Тогда в качестве коэффициентов $\alpha_r(d)$ можно взять характеристические функции

$\chi_r(d)$ элементов разбиения $\mathfrak{D}_r$. В результате дискретного агрегирования получаем мультимодальную систему

$$\bar{X}_- = \chi_1(d)\bar{\mathrm{F}}^1 \circ \bar{X}_+ + \ldots + \chi_S(d)\bar{\mathrm{F}}^S \circ \bar{X}_+. \qquad (3.29)$$

Если $d \in \mathfrak{D}_r$, то эта система совпадает с локальной моделью $\bar{X}_- = \bar{\mathrm{F}}^r \circ \bar{X}_+$. Заметим, что в общем случае разбиение Дирихле-Вороного не всегда хорошо соответствует кластерам. Например, точки одного и того же кластера могут попадать в разные элементы разбиения.

В непрерывном случае для каждого из множеств $\hat{D}_1 \cup \ldots \cup \hat{D}_S \subset \mathbb{R}^{\hat{N}}$ вычислим ковариационную матрицу и обозначим через $\varphi_1, \ldots, \varphi_S$ соответствующие $\hat{N}$-мерные гауссианы с центрами $d_1, \ldots, d_S \subset \mathbb{R}^{\hat{N}}$. Положим

$$\bar{\varphi}_r = \frac{\varphi_r(d)}{\varphi_1(d) + \ldots + \varphi_S(d)}. \qquad (3.30)$$

Функции $\bar{\varphi}_r(d)$ образуют разбиение единицы на пространстве $\mathbb{R}^{\hat{N}}$. Полагая $\alpha_r(d) = \bar{\varphi}_r(d)$ получаем в результате непрерывного агрегирования локальных моделей $\bar{X}_- = \bar{\mathrm{F}}^r \circ \bar{X}_+$ мультимодальную окрестностную систему

$$\bar{X}_- = \bar{\varphi}_1(d)\bar{\mathrm{F}}^1 \circ \bar{X}_+ + \ldots + \bar{\varphi}_S(d)\bar{\mathrm{F}}^S \circ \bar{X}_+. \qquad (3.31)$$

В непрерывной схеме, как было уже сказано в п. 3.1, под термином «гауссиан» понимается плотность многомерного нормального распределения или распределения Коши. При этом плотность распределения Коши для мультимодальной окрестностной системы использовать предпочтительнее, так как нормальное распределение быстро убывает и в случае, когда кластеры данных локальных режимов значительно отделены друг от друга, в области перехода между режимами модель будет плохо определена. Медленное убывание распределения Коши (тяжелый хвост) обеспечивает невырожденность модели в областях между кластерами и, соответственно,

лучше описывает переходы между локальными режимами по сравнению с нормальным распределением.

3.4.5. Агрегирование микролокальных мод

Уравнение окрестностной системы для каждого из узлов можно рассматривать как «микросистему», в которой внешними входами являются все входные переменные узла кроме тех, которые соответствуют управляемым входам. Рассмотрим теперь ситуацию, когда идентифицированы или заданы номинальные режимы всех узлов; соответствующие им моды мы будем называть микролокальными. В этом случае каждый локальный режим системы в целом является набором микролокальных режимов узлов. Вообще говоря, не все формальные наборы микролокальных режимов реализуются как локальные режимы системы в целом.

Если пытаться буквально повторить схему композиции локальных режимов, описанную в предыдущем пункте, то нужно сначала решать задачу о перечислении реализуемых наборов. В действительности, изменяя схему, можно обойтись без решения этой трудной задачи, то есть без информации о локальных модах системы в целом. Пусть $D=(\hat{D},D^V)$ – множество всех кортежей внешних входов и состояний системы во все моменты наблюдения. Для каждого узла $i\in V$ кортежи множества D можно сократить до входов этого узла; полученные наборы редуцированных кортежей обозначим $D(i)\subset\mathbb{R}^{N(i)}$. Далее нужно повторить для каждого узла конструкцию, описанную в предыдущем пункте, и написать соответствующие мультимодальные уравнения, то есть агрегировать микролокальные моды данного узла. Эти мультмодальные уравнения в совокупности задают требуемую агрегированную мультимодальную окрестностную систему. Заметим, что описанная в предыдущем пункте мультимодальная система соответствует кластеризации множества внешних входов $\hat{D}\subset\mathbb{R}^{\hat{N}}$, в то время как

агрегированная полимодальная система соответствует множеству кластеризаций всех наборов кортежей $D(i) \subset \mathbb{R}^{N(i)}$. Таким образом, несмотря на формальное сходство, две описанные схемы, как правило, будут приводить к разным моделям. Выбор наилучшей, при условии знания всех номинальных режимов как системы в целом, так и отдельных подсистем, можно провести классическими статистическими методами.

3.5. Выводы по главе 3

В главе 3 получены следующие результаты:

1. Предложен статический алгоритм рекуррентной иерархической идентификации (R-алгоритм), отличающийся использованием заданной иерархической кластеризации обучающих входных данных и позволяющий повысить точность моделей.

2. Предложен динамический алгоритм рекуррентной иерархической идентификации (L-алгоритм), отличающийся построением в процессе идентификации иерархического разбиения обучающих входных данных и позволяющий улучшать точность моделей и качество прогноза.

3. Предложена численная модификация L-алгоритма на основе трихотомии невязок промежуточных моделей.

4. Предложен метод анализа остаточных данных промежуточных моделей в алгоритмах рекуррентной идентификации, отличающийся введением порядка на множестве обучающих данных и позволяющий контролировать количество уровней иерархии, необходимое для достижения соответствия между точностью и качеством прогноза иерархической модели.

5. Предложены схемы агрегирования локальных мод и микролокальных мод окрестностной системы в мультимодальную окретностную систему.

4. ПРИМЕНЕНИЕ АЛГОРИТМОВ ИЕРАРХИЧЕСКОЙ ИДЕНТИФИКАЦИИ

В данной главе рассматривается применение алгоритмов рекуррентной иерархической идентификации с анализом остаточных данных к задачам прогнозирования характеристик клинкера и прогнозирования температурного режима стадии диффузии производства сахара.

Представлены примеры иерархической идентификации двумя разработанными алгоритмами на множестве сгенерированных обучающих данных. Показано сравнение на сгенерированном множестве данных итоговых показателей работы L-алгоритма (для 2-уровневого и 3-уровнего дерева T) с другими известными методами идентификации, такими как, ансамбли деревьев регрессии (бустинг и случайный лес) и нейронные сети.

Идентификация моделей прогнозирования характеристик клинкера и температурного режима диффузии сахара производилась согласно L-алгоритму, в том числе с использованием метода трихотомии невязок. Также производилось сравнение с моделями, идентифицированными с помощью методов бустинга и случайного леса деревьев регрессии и нейросетей.

4.1. Примеры иерархической идентификации с анализом остаточных данных

В первом примере рассматривается задача статической рекуррентной идентификации (R-алгоритм) иерархической окрестностной модели на основе иерархической кластеризации. Функционирование R-алгоритма описывалось в п. 3.1. Данный алгоритм предполагает наличие априорного задания дерева модели, которое основано на иерархической кластеризации пространства входных данных. В связи с этим уместно применять R-алгоритм в ситуациях, когда датасет (с учетом или без учета значений таргетов) имеет явно выраженное изначальное разбиение или определенную структуру входных

данных, в том числе и иерархическую. Такие структуры часто возникают у переменных, например, подверженных сезонности – годовой, квартальной, месячной, недельной, или у параметров каких-либо неоднородных процессов и систем, распределенных в пространстве и/или во времени.

Данные изначальные разбиения существенно облегчают возможность построения дерева модели, так как заранее известно сколько необходимо построить кластеров на каждом уровне иерархии и определить глубину дерева. При этом необходимо помнить, что у слишком глубоко построенного дерева (в случаях, если модель сложна или переобучена) можно редуцировать концевые вершины, в том числе несколько раз на одной ветке дерева.

В качестве примера иерархической идентификации по R-алгоритму рассмотрим сгенерированное множество обучающих данных с двумерной областью определения для входов в виде квадрата $[0{,}1; 1{,}9] \times [0{,}1; 1{,}9]$ и одномерного выхода по следующему алгоритму (см. в [111]):

1) Датчиком равномерно распределенных случайных чисел на [0; 1] генерируется выборка объема N=400, по 200 значений для каждой из двух переменных;

2) Значения переменных умножаются на k=0,8, тем самым случайные числа «стягиваются» к промежутку [0; 0,8];

3) Прибавляются к первым 100 значениям и ко вторым 100 значениям обеих переменных x_1 и x_2 соответственно числа 0,1 и 1,1. «Стянутые» с помощью коэффициента k=0,8 значения, таким образом, «центрируются» на промежутках [0,1; 0,9] и [1,1; 1,9], что позволяет искусственно создать 4 отдельных кластера на плоскости;

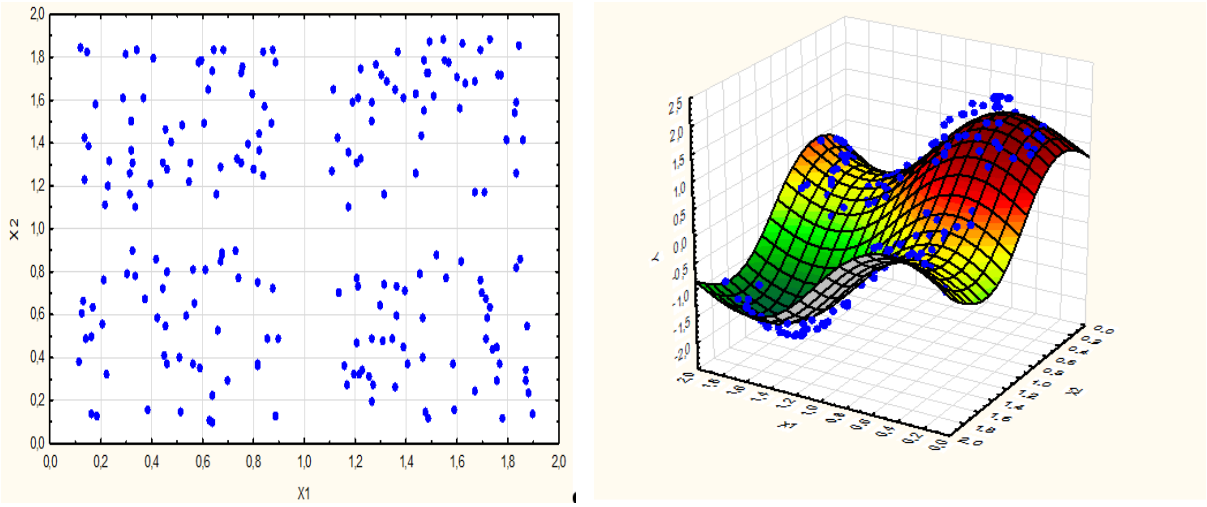
4) «Стянутые» и «центрированные» первые и вторые 100 значений обеих переменных и располагаются в случайном порядке. В результате получается случайная выборка с 200 измерениями входной величины $X = (x_1, x_2) \in \mathbb{R}^2$, расположенных в 4-х «кластерных» квадратах $[0{,}1; 0{,}9] \times [0{,}1; 0{,}9]$, $[0{,}1; 0{,}9] \times [1{,}1; 1{,}9]$, $[1{,}1; 1{,}9] \times [0{,}1; 0{,}9]$ и $[1{,}1; 1{,}9] \times [1{,}1; 1{,}9]$;

5) Для генерации выходной переменной $Y \in \mathbb{R}^1$ выбирается какая-либо нелинейная функция, учитывая, что реальные процессы также могут быть существенно нелинейны и хаотичны. Для данного примера генерируется выход (таргет) с помощью функции $\hat{y} = sin\pi x_1 + sin\pi x_2$.

В случае нелинейной, но детерминированной функции при генерации выхода можно будет оценивать только качество аппроксимации. Для дополнительного контроля структуры слагаемых модели вместо произвольной нелинейной функции $\hat{y} = f(x_1, x_2)$ можно использовать, например, функцию $L_i(x_1, x_2) = ax_1 + bx_2 + c + w_i(x_1, x_2)(a_i x_1 + b_i x_2 + c_i)$.

По заданным формулам для каждой из четырех пар (i = 1, 2, 3, 4) выборок вычисляются значения выходов. Здесь $w_i(x_1, x_2) \in (1 - d, 1 + d)$, $d \in (0,1)$ – равномерно распределенная на интервале $(1 - d, 1 + d)$ случайная величина (шум). В этом случае можно сравнивать ожидаемые формулы для общей линейной части и корректирующих слагаемых с формулами, полученными в результате дальнейшей идентификации.

Диаграмма рассеяния двух переменных x_1 и x_2, а также совместного 3D-графика всех 3-х переменных (с пакетной подгонкой поверхности) представлены на рисунке 4.1, соответственно (а) и (б).



а) б)

Рисунок 4.1 – Диаграмма рассеяния сгенерированных значений:

а) x_1 и x_2, б) x_1, x_2 и $\hat{y}$

До начала работы R-алгоритма формируется дерево модели *T*. В данном примере на множестве сгенерированных данных, входная величина $X = (x_1, x_2)$ была сгенерирована так, чтобы были явно выделены 4 области-кластера (рисунок 4.1, а). Так как дальнейшего разбиения в каждой из этих 4-х областей визуально не наблюдается, дерево модели задается двухуровневым, где первый уровень является корневым узлом, а на втором уровне расположены 4 концевых вершины, соответствующие заданию 4-х кластеров в математическом пакете, имеющем модуль «кластерный анализ». Кластеризация производилась по методу *K*-средних в пакете Statistica. Получены следующие результаты:

– кластер 1 содержит 48 значений величины $X = (x_1, x_2)$ со средним значением (центром кластера) в точке (1,479; 1,584);

– кластер 2 содержит 52 значения наблюдений с центром в точке (1,506; 0,484);

– кластер 3 содержит 52 значения наблюдений с центром в точке (0,516; 1,491);

– кластер 4 содержит 48 значений наблюдений с центром в точке (0,496; 0,537).

Центры кластеров и количество значений $X = (x_1, x_2)$ в каждом из них соответствует алгоритму генерации равномерно распределенной случайной величины, ее сжатия и сдвигов.

Далее, согласно R-алгоритму, идентифицируется начальная, а в данном случае линейная, модель множественной регрессии по всем данным. Она имеет вид:

$$F^1(x) = 2{,}42256 - 1{,}18135x_1 - 1{,}22023x_2. \qquad (4.1)$$

Для данной начальной линейной модели множественный коэффициент корреляции R = 0,875, коэффициент детерминации R^2 = 0,7656, а сумма квадратов ошибок равна $\sum e^2 = 54{,}1792$.

На втором шаге вычисляются невязки начальной модели, которые ставятся в соответствие множествам входов каждого кластера. Далее для множеств входных данных каждого кластера и соответствующих им невязок строятся уточняющие, в данном случае вновь линейные, модели (второго уровня иерархии) $F^2_{i(x)}(x)$ ($i(x) = 1, 2, 3, 4$ – номер кластера).

Линейные модели множественной регрессии величины невязок e по факторам $X = (x_1, x_2)$ в четырех кластерах имеют вид:

$$\begin{aligned} F^2_1(x) &= -3{,}72597 + 1{,}24896x_1 + 1{,}01925x_2 \\ F^2_2(x) &= -2{,}34369 + 1{,}18929x_1 + 1{,}09809x_2 \\ F^2_3(x) &= -2{,}35356 + 0{,}98856x_1 + 1{,}23357x_2 \\ F^2_4(x) &= -0{,}96138 + 1{,}13729x_1 + 1{,}28349x_2 \end{aligned} \qquad (4.2)$$

По итогу строится двухуровневая квазилинейная модель с иерархией кластеров входных данных:

$$F(x) = F^1(x) + F^2_{i(x)}(x), \qquad (4.3)$$

где $i(x)$ – выбор кластера и соответствующей модели второго уровня в зависимости от (x_1, x_2).

Для данной иерархической квазилинейной модели множественный коэффициент корреляции R=0,966, коэффициент детерминации R^2=0,933, а сумма квадратов ошибок равна $\sum e^2 = 15{,}5$. Сумма квадратов ошибок снизилась в 3,5 раза или на 71,4 % относительно исходной линейной модели.

На рисунке 4.2 изображены отдельные значения сгенерированных выходов (синие точки) и значения двухуровневой модели (красные соединенные линии). Визуально видно, что аппроксимация иерархической моделью проходит «внутри» каждого кластера, что существенно влияет на ее качество.

Прогнозирование по данной модели осуществляется следующим образом: в пространстве входов ищется ближайшая к прогнозируемому кортежу точка из кластеризованного обучающего множества, добавочная модель второго уровня $F^2_{i(x)}(x)$ «включается» в тот момент, когда данный

кортеж попадает в тот или иной кластер (лист дерева), необходимая добавочная модель суммируется с моделями предыдущих уровней при движении по дереву модели «снизу-вверх».

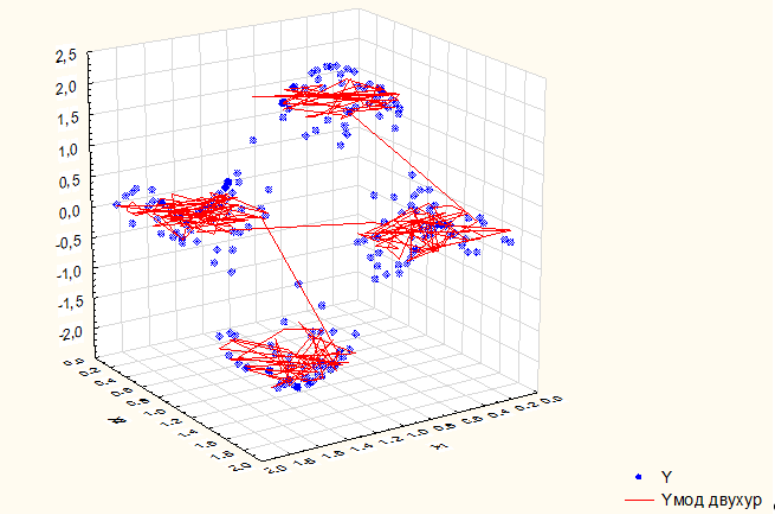


Рисунок 4.2 – Рассеяние значений выходов и их аппроксимация иерархической окрестностной моделью

Во втором примере рассматривается задача динамической рекуррентной идентификации (L-алгоритм) иерархической окрестностной модели на основе иерархического разбиения. Функционирование L-алгоритма описывалось в п. 3.2 третьей главы. Также во втором примере, помимо идентификации иерархической окрестностной модели, сравним ее итоговые показатели с моделями на основе известных алгоритмов, такими, как ансамбли деревьев регрессии и нейронными сетями.

Идентификация модели с помощью L-алгоритма на множестве сгенерированных данных подробно рассматривалась в работе [171]. В данном примере рассмотрена похожая схема генерирования и идентификации.

Датчиком равномерно распределенных случайных чисел на отрезке $[0; 1]$ генерируется 450 значений двумерного входа $X = (x_1, x_2)$, $\hat{y}$ – одномерный выход, является нелинейной функцией факторов вида $\hat{y} = \sin(20x_1) + \sin(20x_2)$. Дополнительно для целей сравнения различных моделей на прогнозировании генерируется дополнительно 50 значений входа

и выхода. Диаграмма рассеяния трех переменных и подгонка графика значений $\hat{y}$ показаны на рисунке 4.3, а.

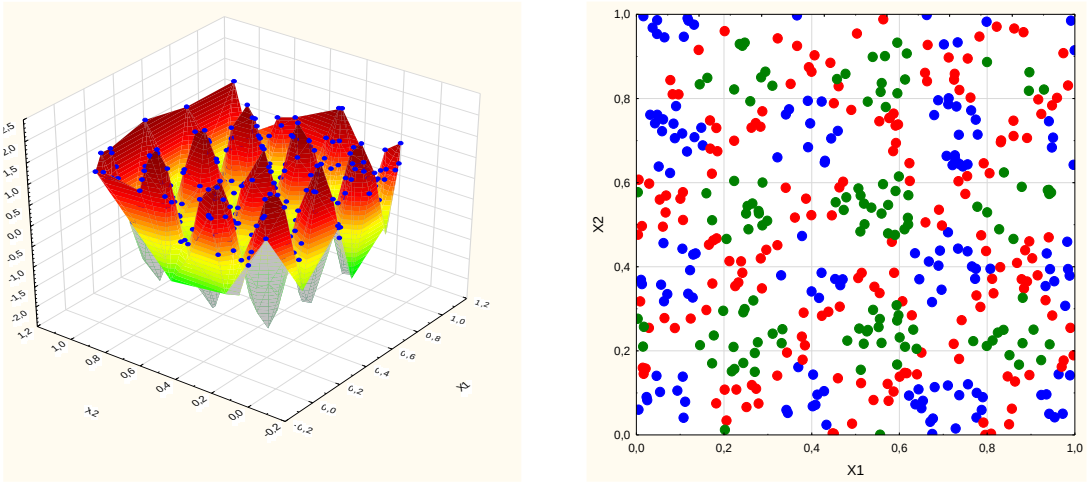
Для моделей бустинга деревьев, случайного леса и нейронных сетей, подлежащих не только обучению, но и тестированию, датасет из 450 сгенерированных данных был разбит на два подмножества: 90 % (405 измерений) для обучения и 10 % (45 измерений) для тестирования. Выбор подмножеств, как и идентификация известных в литературе моделей, осуществляется в математическом пакете Statistica автоматически. Идентификация иерархических окрестностных моделей производилась с использованием пакета Statistica и Matlab.

Сравнения моделей как при обучении с тестированием, так и при предсказании производилось с помощью суммы квадратов остатков (SE) и среднеквадратичной ошибки (RMSE, СКО). Относительная ошибка аппроксимации для близких к нулю значений $\hat{y}$ будет работать очень плохо, поэтому ее для сравнения моделей в этом примере предлагается не использовать.

В соответствии с L-алгоритмом, идентифицируется начальная, в данном случае линейная, модель множественной регрессии по всем данным. Она имеет вид:

$$F^1(x) = 0{,}318546 - 0{,}309985x_1 - 0{,}180686x_2. \qquad (4.4)$$

На следующем шаге алгоритма проводится кластеризация остатков начальной линейной модели по 3 кластерам. Выбор количества кластеров определяется исследователем или задачами исследования. Сопоставляя кластеризованным значениям остатков их прообразы, то есть соответствующие значения входной величины $X = (x_1, x_2)$, получаем иерархическое разбиение входов. Это разбиение представлено на рисунке 4.3, б. Точки рассеяния входа для 1-го разбиения показаны синим цветом, для 2-го разбиения – красным цветом, для 3-го разбиения – зеленым цветом.



а) б)

Рисунок 4.3 – Диаграмма рассеяния обучающего множества с подгонкой поверхности (а) и разбиение пространства входов на 3 подмножества (б)

Одновременно с процессом кластеризации остатков, т.е. динамически, строится дерево модели T: от корневого узла идут направленные дуги к трем узлам второго уровня, соответствующих трем кластерам остатков и разбиений множества входных данных. Для каждого узла, т.е. для каждого из разбиений, на следующем шаге алгоритма строятся уточняющие модели, в данном примере – линейные.

$$F_1^2(x) = 1{,}17 - 0{,}0598x_1 - 0{,}0111x_2$$
$$F_2^2(x) = -0{,}15422 + 0{,}11757x_1 + 0{,}0563878x_2 \qquad (4.5)$$
$$F_3^2(x) = -1{,}35282 + 0{,}1235132x_1 + 0{,}046475x_2$$

Двухуровневая квазилинейная окрестностная модель с иерархией разбиения входных данных, основанных на кластерах невязок линейной модели первого уровня:

$$F(x) = F^1(x) + F^2_{i(x)}(x), \qquad (4.6)$$

где i(x) – выбор кластера и соответствующей модели второго уровня в зависимости от (x_1, x_2). На данном шаге, в зависимости от анализа остаточных данных, процесс иерархической идентификации можно остановить или продолжить. В данном примере строится трехуровневая модель. Каждое из 3-х разбиений пространства входов, аналогично предыдущим шагам,

разбивается на 2 подмножества. Дерево модели T, таким образом, состоит из 6 листьев. Линейные уточняющие модели каждого листа дерева имеют вид:

$$F_{11}^{3}(x) = -0{,}204278 - 0{,}03244x_1 - 0{,}0397x_2$$
$$F_{12}^{3}(x) = 0{,}286 + 0{,}091x_1 + 0{,}1021383x_2$$
$$F_{21}^{3}(x) = 0{,}30434 - 0{,}0443444x_1 - 0{,}0702x_2 \quad (4.7)$$
$$F_{22}^{3}(x) = -0{,}17168 - 0{,}13275x_1 - 0{,}051x_2$$
$$F_{31}^{3}(x) = -0{,}46947 - 0{,}0634222x_1 + 0{,}14515x_2$$
$$F_{32}^{3}(x) = 0{,}342426 + 0{,}0298x_1 - 0{,}1194815x_2$$

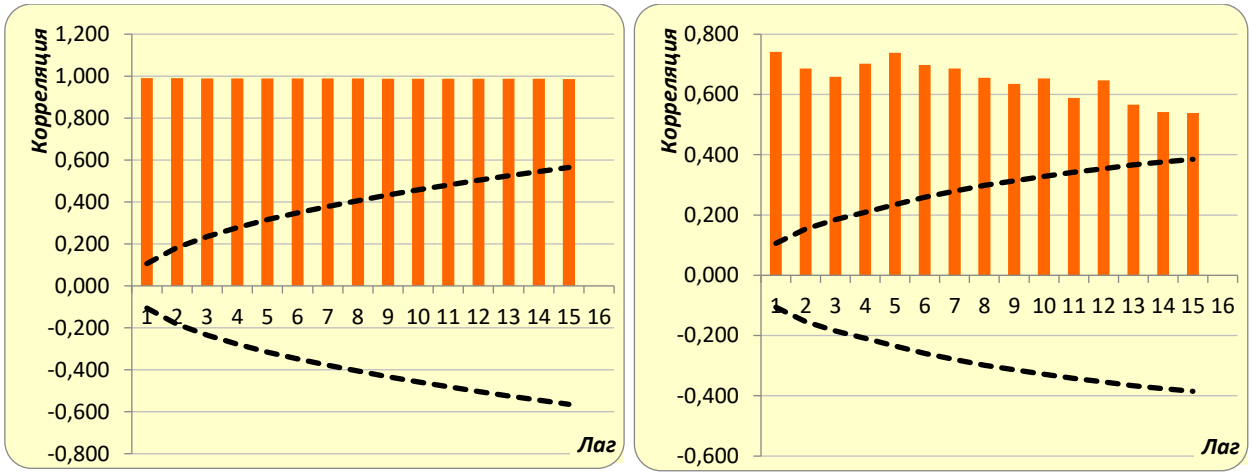
Трехуровневая квазилинейная окрестностная модель:

$$F(x) = F^{1}(x) + F_{i(x)}^{2}(x) + F_{ji(x)}^{3}, \quad (4.8)$$

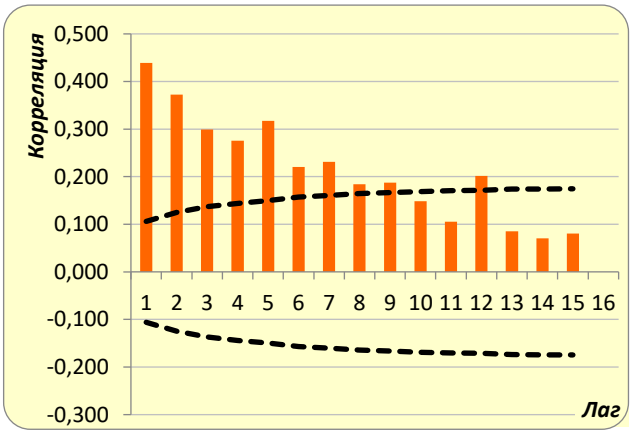
где $ji(x)$ – выбор кластера и соответствующей модели третьего уровня в зависимости от (x_1, x_2).

Как отмечалось в п. 3.3.2 третьей главы, для остановки работы алгоритмов иерархической идентификации предлагается рассчитывать АКФ остатков многоуровневых моделей, упорядоченных в соответствии с возрастанием (или убыванием) входной переменной. Достижение значений АКФ остатков общей многоуровневой модели на данном шаге алгоритма, близкой к АКФ для «белого шума», предполагает достижения ситуации, при которой ошибка модели распределена стохастично и больше не содержит качественной информации, модель уже учитывает всю необходимою информацию о моделируемом процессе. Дальнейшая идентификация таких стохастично распределенных остатков приведет к нежелательным последствиям, аналогичных эффектам переобучения в теории решающих деревьев и нейронных сетей. Проводится анализ АКФ остатков с помощью анализа коррелограммы АКФ. На рисунке 4.4 представлены коррелограммы АКФ остатков, упорядоченных в соответствии с возрастанием сгенерированного значения выхода $\hat{y}$, соответственно для начальной линейной модели (а), двухуровневой квазилинейной модели (б) и трехуровневой квазилинейной модели (в).

Анализ представленных на рисунке 4.4 коррелограмм показывает, что с ростом уровней иерархий автокорреляция остатков для всех лагов снижается (следует обратить внимание на вертикальную шкалу корреляций). Однако АКФ остатков трехуровневой модели (рисунок 4.4, в) все еще не близка к АКФ для «белого шума», что говорит о возможности дальнейшей работы рекуррентного L-алгоритма, «углубления» дерева модели T и идентификации дополнительных линейных слагаемых. В рамках данного примера остановка производится на идентификации трехуровневой общей окрестностной модели.



а) б)



в)

Рисунок 4.4 – Коррелограммы АКФ остатков иерархических моделей:
а) исходной линейной, б) двухуровневой, в) трехуровневой

В результате обучения различных автоматизированных нейросетевых моделей (NN) в пакете Statistica, были выбраны 2 с наименьшей суммой квадратов ошибки, а именно:

1) многослойный перцептрон с 8 скрытыми нейронами (MLP 2-8-1), обучавшийся по алгоритму BFGS с 8610 эпохами обучения, с функциями активации скрытых нейронов – гиперболической, и выходного нейрона – гиперболической.

2) многослойный перцептрон с 6 скрытыми нейронами (MLP 2-6-1), обучавшийся по алгоритму BFGS с 10000 эпохами обучения, с функциями активации скрытых нейронов – логистической, и выходного нейрона – логистической.

Результаты идентификации всех сравниваемых моделей представлены в таблице 4.1.

Таблица 4.1

Результаты идентификации моделей по сгенерированным данным

Бустинг деревьев		Случайный лес		NN MLP 2-8-1		NN MLP 2-6-1	
SE	RMSE	SE	RMSE	SE	RMSE	SE	RMSE
138,8	0,555	117,38	0,51	147,8	0,574	155,42	0,592

L-модель 2-уровн.		L-модель 3-уровн.	
SE	RMSE	SE	RMSE
55,41	0,346	14,32	0,173

Как видно из таблицы 4.1, квадратичная ошибка (SE) иерархических окрестностных моделей на порядки ниже, чем у известных алгоритмов моделей регрессии. В частности, наилучшая по показателю SE модель случайного леса деревьев регрессии в 8,2 раза хуже аппроксимирует данные, чем трехуровневая квазилинейная окрестностная модель, или более чем

на 700 %. Отмечается и снижение среднеквадратичной ошибки (RMSE) для иерархических окрестностных моделей. Формулы для расчетов показателей SE и RMSE:

$$SE = \sum_{i=1}^{n} e_i^2. \tag{4.9}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n} e_i^2}, \tag{4.10}$$

где $e_i = y_i - \hat{y}_i$ $(i = 1,\ldots, n)$ – разница между фактическими и модельными значениями целевой величины (ошибка модели), n – объем выборки обучающих данных.

Далее сравнивается квадратичная ошибка для прогнозной выборки из сгенерированных данных объема 50 измерений. Сравнения идентифицированных моделей на прогнозируемой выборке представлены в таблице 4.2.

Таблица 4.2

Результаты прогнозирования моделей по сгенерированным данным

Бустинг деревьев		Случайный лес		MLP 2-8-1		MLP 2-6-1	
SE	RMSE	SE	RMSE	SE	RMSE	SE	RMSE
12,1535	0,493	14,5308	0,539	20,9897	0,648	21,794	0,66

L-модель 2-уровн.		L-модель 3-уровн.	
SE	RMSE	SE	RMSE
11,2625	0,475	6,7532	0,368

Как видно из таблицы 4.2, квадратичная ошибка двухуровневой окрестностной модели для прогнозной выборки сопоставима с наилучшей из известных моделей, в данном случае алгоритмами бустинга над деревьями регрессии. Однако уже трехуровневая модель, идентифицированная по

L-алгоритму, имеет квадратичную ошибку, в 1,8 раза меньшую, чем у бустинга деревьев регрессии, что в процентной разности составляет примерно 80 %. RMSE трехуровневой L-модели также лучше, чем у модели бустинга.

4.2. Задача прогнозирования характеристик и качества клинкера

Цементное производство – многоэтапный сложный процесс, при реализации которого главным компонентом в получении готового продукта является клинкер. Клинкер получают путем обжига до спекания тонкодисперсной сырьевой муки, состоящей из известняка, глины, шлакового щебня и других материалов (см. [1, 24]).

Обжиг сырьевой смеси осуществляется во вращающейся печи, представляющей собой пустотелый, открытый с торцов барабан, вращающийся со скоростью 1-1,5 об/мин в зависимости от диаметра и производительности печи. Сырье в виде порошка или шлама подается в печь с верхней холодной части, а со стороны нижнего конца (горячая часть) вдувается топливо (природный газ, мазут, воздушно-угольная смесь). Сырье медленно движется к нижнему концу навстречу горячим газам, проходя различные температурные зоны. Условно вращающуюся печь можно разделить на шесть температурных зон в зависимости от характера протекающих в них процессов.

В зоне сушки (испарения) происходит высушивание поступившей сырьевой смеси при постепенном повышении температуры от 70 до 200 ℃. Далее подсушенный материал подается в зону подогрева, где при постепенном нагревании сырья с 200 до 700 ℃ сгорают находившиеся в нем органические примеси. В зоне декарбонизации температура обжигаемого материала поднимается до 1100 ℃ и завершается процесс диссоциации (разделения на молекулярные элементы). В зоне экзотермических реакций (1100-1250 ℃) проходят твердофазовые реакции, сопровождающиеся выделением большого количества теплоты и интенсивным повышением температуры материала. В

зоне спекания температура достигает наивысшего значения (1450 ℃), необходимого для частичного плавления материала и образования главного минерала клинкера – алита. В зоне охлаждения температура клинкера понижается до 1000 ℃. После обжига в печи клинкер необходимо интенсивно охладить до 100-200 ℃ в специальных холодильниках с помощью холодных воздушных потоков. После процессов обжига и охлаждения клинкер хранится на складе одну-две недели для окончательной кристаллизации и гашения окиси кальция за счет естественной влаги из воздуха.

В конечном виде клинкер представляет собой мелкие камнеподобные зерна – гранулы темно-серого или зеленовато-серого цвета. По микроструктуре клинкер представляет собой сложную тонкозернистую смесь кристаллических фаз и небольшого количества стекловидной фазы.

На процесс клинкерообразования влияет ряд следующих факторов:

- минералогический состав. Существенно влияет на спекаемость клинкера. Получение данного компонента цемента в значительной степени зависит от тонкости помола сырьевой муки и равномерности ее зернового состава;

- температура обжига. При высоких температурах ускоряется синтез составляющих клинкера, резко снижается вязкость жидкой фазы и увеличивается ее количество, что оказывает решающее влияние на процесс получения клинкера;

- минерализаторы. Введение в сырьевую смесь дополнительных веществ приводит к увеличению количества расплава, изменению его свойств, а также создает благоприятные условия для расплавления и кристаллизации компонентов состава;

- химический состав. Оказывает существенное влияние на скорость реакций, протекающих при обжиге, и предопределяет длительность процесса в целом. Именно этот фактор, а именно так называемые модульные характеристики, получаемые при изучении химического состава, позволяют

при помощи математической модели оценить качество клинкера еще до процесса обжига.

Формулы для расчета модульных характеристик клинкера представлены ниже.

1) Силикатный модуль:

$$n = \frac{SiO_2\%}{Al_2O_3\% + Fe_2O_3\%}. \quad (4.11)$$

При высоком значении *n* сырьевая смесь трудно обжигается, вследствие чего необходима более высокая температура обжига. При пониженном же значении силикатного модуля возрастает количество жидкой фазы в обжигаемом материале и обжиг проходит легче, однако получаемый клинкер характеризуется медленным твердением и пониженной прочностью.

2) Глиноземный (алюминатный) модуль:

$$p = \frac{SiO_2\%}{Fe_2O_3\%}. \quad (4.12)$$

При высоком значении глиноземного модуля цементы обладают пониженной коррозионной стойкостью. При пониженном значении модуля повышается вязкость расплава, что затрудняет обжиг.

3) Коэффициент насыщения:

$$KH = \frac{CaO\% - 1,65 \cdot Al_2O_3\% - 0,35 \cdot Fe_2O_3\%}{2,8 \cdot SiO_2\%}. \quad (4.13)$$

При высоком *КН* клинкер трудно обжигается, в связи с чем цемент быстро твердеет и обладает высокой прочностью, но пониженной коррозионной стойкостью. Пониженный коэффициент насыщения приводит к тому, что без дополнительных мер цемент очень медленно твердеет и не пригоден для использования в обычных условиях, цементный камень обладает повышенной коррозостойкостью и используется как составляющая тампонажных цементов.

Немаловажным фактором на этапе производства клинкера также является скорость охлаждения, в связи с тем, что от нее зависит соотношение

содержащихся в клинкере кристаллической и стекловидной фаз, т.е. формирующаяся структура, а также минералогический состав и размолоспособность, что в свою очередь влияет на качество цемента. При медленном охлаждении клинкера стекловидная фаза почти полностью кристаллизуется, в результате чего увеличивается содержание свободной окиси кальция (жженая известь), способного вызвать растрескивание готового продукта. Цементы на основе таких клинкеров, как правило, отличаются пониженной активностью и медленно твердеют. При быстром охлаждении в клинкерах наблюдается повышенное содержание стекловидной фазы и мелкокристаллическая структура. Эти факторы приводят к более легкому размалыванию клинкера. Быстроохлаждаемые клинкеры обеспечивают цементам более высокую активность, повышенную сульфатостойкость и пониженную усадку. Рекомендуется медленно охлаждать клинкер до 1200 ℃ в печи с последующим быстрым охлаждением до нормальной температуры в холодильниках. Наиболее резкому охлаждению следует подвергать клинкеры с повышенными значениями глиноземного и силикатного модулей.

Таким образом, между спекаемостью клинкера, коэффициентом насыщения (*КН)* и модулями *n, p* существует непосредственная связь, при этом свойства клинкера (в виде его модульных характеристик) сильно влияют на качество конечного цементного продукта. Важным представляется разработка математической модели прогнозирования данных характеристик на основе измерений химического состава сырьевой муки до обжига в печах. Например, в работе [123], прогнозировались значения извести *CaO* клинкера на основе параметров вращающейся печи с применением методов машинного обучения.

Анализируя процесс клинкерообразования, а также прогнозируя получаемые свойства материала с помощью иерархических окрестностных моделей и рекуррентных алгоритмов, можно будет делать определенные выводы и принимать решения по управлению параметров печи, что, в свою

очередь, приведет к повышению скорости, качества, стабильности и в целом совершенствованию процесса производства цемента.

4.3. Иерархические модели прогнозирования характеристик клинкера

В работах [112, 171] представлены результаты применения динамического L-алгоритма рекуррентной идентификации к задаче построения иерархической окрестностной модели прогнозирования значений глиноземного модуля клинкера после обжига сырьевой муки во вращающихся печах. В этом пункте рассмотрим применение динамического L-алгоритма идентификации, в одном случае с кластеризацией невязок, и во втором случае с трихотомией невязок, к задаче построения иерархических моделей прогнозирования всех трех модульных характеристик клинкера, а также моделирование модульных характеристик традиционными линейной и квадратичной регрессией, алгоритмами случайного леса и бустинга деревьев регрессии и нейронных сетей. Последние методы и алгоритмы применяются с целью сравнения результатов всех идентифицируемых моделей.

Разработанный комплекс программ позволяет осуществить процедуру идентификации иерархических моделей. Идентификация и прогнозирование разработанных моделей осуществлялись с применением пакета Statistica и среды Matlab. Выбор ближайших точек для дерева иерархических окрестностных моделей в задачах прогнозирования производился с помощью функции в среде Matlab. Также использовалась программа в среде Mathcad (см. [105]) для анализа исходных данных.

Для моделирования использовались 600 измерений химического состава сырьевой муки в качестве входов и модульных характеристик клинкера после обжига во вращающихся печах в качестве одномерного выхода.

По результатам L-алгоритма идентификации с кластеризацией невязок модели прогнозирования глиноземного модуля p дерево модели содержит 3 узла на втором уровне и 10 листьев на третьем уровне, так как в процессе

построения дерева модели один из узлов второго уровня содержал существенно больше элементов подмножества обучающего множества, чем в двух других узлах, в результате чего было принято решение для данного узла построить 4 выходящие дуги. Для L-алгоритма с трихотомией невязок дерево модели содержит 3 узла на втором уровне (один из которых является листом), 6 узлов на третьем уровне (два из которых являются листьями) и 12 листьев на четвертом уровне.

Оценки значений средней квадратичной ошибки и средней ошибки аппроксимации в процентах всех идентифицируемых моделей глиноземного модуля клинкера p представлены в таблице 4.3.

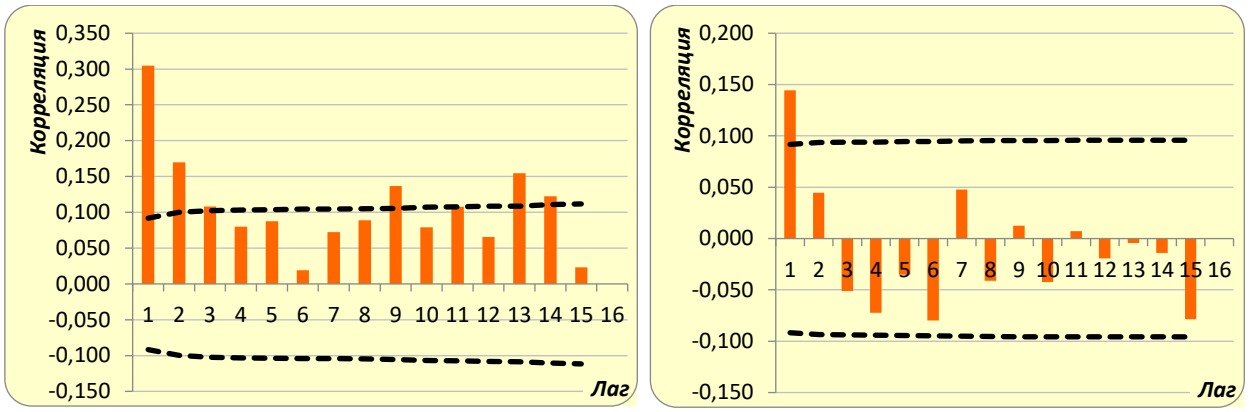
Таблица 4.3

Результаты идентификации различных моделей аппроксимации глиноземного модуля p

Модель	Среднеквадратичная ошибка (RMSE) и средняя ошибки аппроксимации ($\bar{A}$) глиноземного модуля p	
	RMSE	$\bar{A}$, %
Линейная регрессия	0,0478	3,11
Квадратичная регрессия	0,047	3,05
Бустинг деревьев	0,046	2,92
Случайный лес	0,0464	2,98
Нейросеть	0,04431	2,84
L-модель 2-уровневая с кластеризацией	0,021	1,33
L-модель 3-уровневая с кластеризацией	0,0075	0,46
L-модель 3-уровневая с трихотомией	0,012	0,68
L-модель 4-уровневая с трихотомией	0,0068	0,42

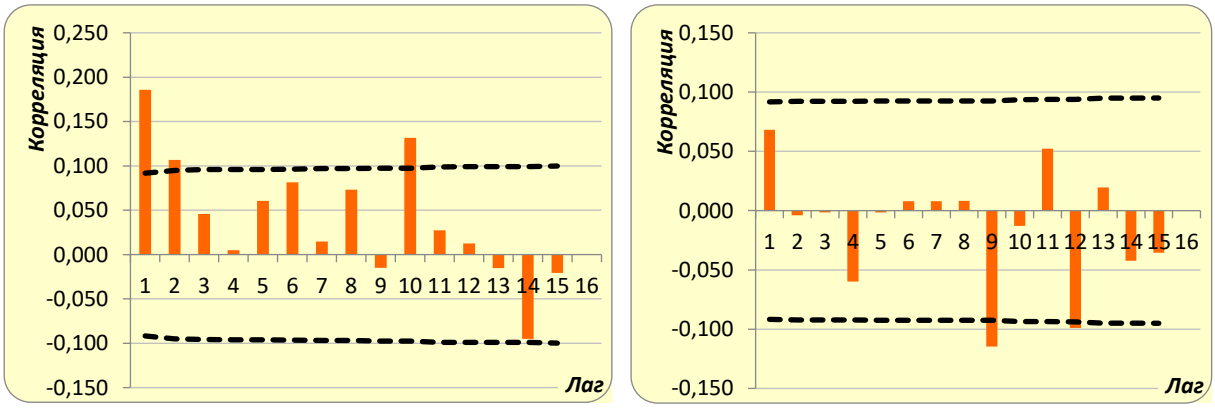
Как видно, лучшие показатели аппроксимации у трехуровневой окрестностной модели, построенной по L-алгоритму с кластеризацией невязок, и четырехуровневой окрестностной модели, построенной по L-алгоритму с трихотомией невязок. Четырехуровневая модель,

идентифицированная по L-алгоритму с трихотомией невязок, имеет среднюю ошибку аппроксимации в 6,76 раза меньшую, чем модель автоматизированной нейронной сети, что в процентной разности составляет примерно 576 %. Процент улучшения аппроксимации, согласно этому же показателю, четырехуровневой модели на основе трихотомии по сравнению с трехуровневой моделью на основе кластеризации, составляет 9,5 %.



а) б)

Рисунок 4.5 – Коррелограммы АКФ остатков иерархических моделей, идентифицированных по L-алгоритму с кластеризацией остатков: а) двухуровневой, б) трехуровневой



а) б)

Рисунок 4.6 – Коррелограммы АКФ остатков иерархических моделей, идентифицированных по L-алгоритму с трихотомией остатков: а) трехуровневой, б) четырехуровневой

Остановка алгоритмов идентификации моделей глиноземного модуля, согласно описанному методу в п. 3.4, происходила на основе анализа АКФ невязок, коррелограммы которых представлены на рисунках 4.5-4.6.

Результаты моделирования силикатного модуля *n* следующие: дерево модели L-алгоритма идентификации с кластеризацией невязок содержит 3 узла на втором уровне и 9 листьев на третьем уровне; дерево модели L-алгоритма с трихотомией невязок по структуре аналогично дереву модели глиноземного модуля.

Оценки значений среднеквадратичной ошибки и средней ошибки аппроксимации в процентах всех идентифицируемых моделей силикатного модуля клинкера *n* представлены в таблице 4.4.

Таблица 4.4

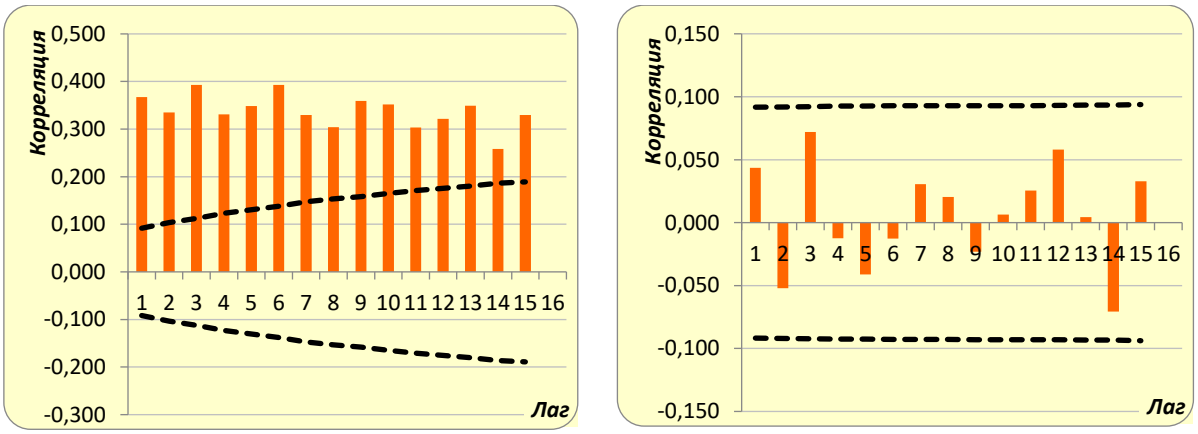
Результаты идентификации различных моделей аппроксимации силикатного модуля *n*

Модель	Среднеквадратичная ошибка (RMSE) и средняя ошибки аппроксимации ($\bar{A}$) силикатного модуля *n*	
	RMSE	$\bar{A}$, %
Линейная регрессия	0,05	1,64
Квадратичная регрессия	0,049	1,61
Бустинг деревьев	0,048	1,56
Случайный лес	0,05	1,61
Нейросеть	0,0442	1,43
L-модель 2-уровневая с кластеризацией	0,0251	1,02
L-модель 3-уровневая с кластеризацией	0,0085	0,28
L-модель 3-уровневая с трихотомией	0,013	0,37
L-модель 4-уровневая с трихотомией	0,0068	0,22

Четырехуровневая модель, идентифицированная по L-алгоритму с трихотомией невязок, имеет среднюю ошибку аппроксимации в 6,5 раза

меньшую, чем модель автоматизированной нейронной сети, что в процентной разности составляет 550 %. Процент улучшения аппроксимации, согласно этому же показателю, четырехуровневой модели на основе трихотомии по сравнению с трехуровневой моделью на основе кластеризации, составляет 27 %.

На рисунке 4.7 показаны коррелограммы АКФ (для лагов 1,…,15) для остатков первоначальной линейной модели регрессии (а) и четырехуровневой модели с трихотомией, упорядоченных в соответствии с возрастанием выходной величины силикатного модуля *n*. Видно, что АКФ остатков становится мало отличимой от АКФ для «белого шума», что является показателем качества модели и остановки работы алгоритма после построения 4-х уровней дерева *T*. Аналогично решалась задача остановки алгоритма с кластеризацией невязок при идентификации трехуровневой модели.



а) б)

Рисунок 4.7 – Коррелограммы АКФ остатков: а) линейной модели; б) четырехуровневой модели, идентифицированной по L-алгоритму с трихотомией невязок

Результаты моделирования коэффициента насыщения *КН,* следующие: дерево модели L-алгоритма идентификации с кластеризацией невязок и дерево модели L-алгоритма с трихотомией невязок по своим структурам аналогичны деревьям моделей силикатного модуля *n*.

Оценки значений средней квадратичной ошибки и средней ошибки аппроксимации в процентах всех идентифицируемых моделей коэффициента насыщения клинкера *КН* представлены в таблице 4.5.

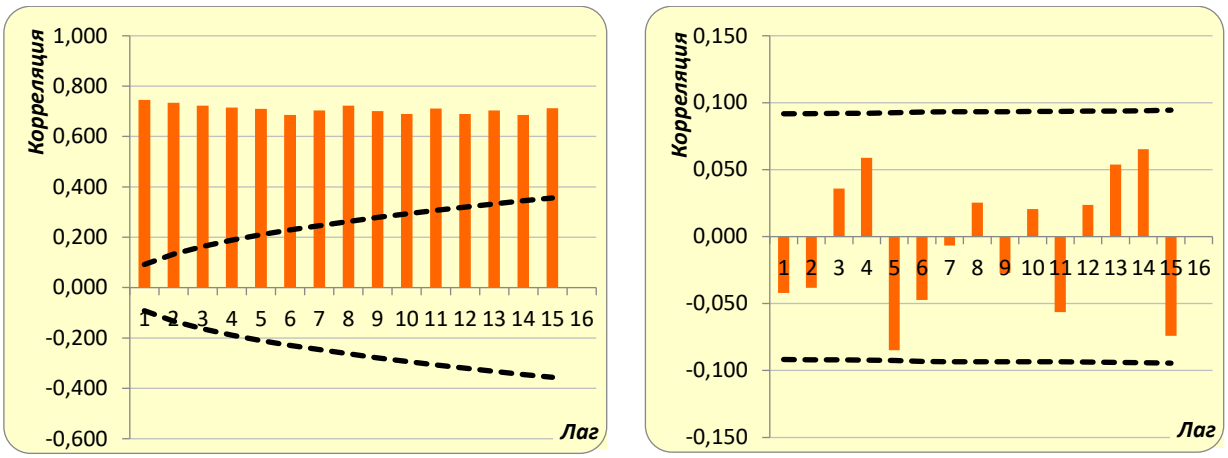
Таблица 4.5

Результаты идентификации различных моделей аппроксимации коэффициента насыщения *КН*

Модель	Среднеквадратичная ошибка (RMSE) и средняя ошибки аппроксимации ($\bar{A}$) глиноземного модуля p	
	RMSE	$\bar{A}$, %
Линейная регрессия	0,0172	1,39
Квадратичная регрессия	0,01717	1,39
Бустинг деревьев	0,01183	0,98
Случайный лес	0,0153	1,14
Нейросеть	0,0096	0,81
L-модель 2-уровневая с кластеризацией	0,0094	0,75
L-модель 3-уровневая с кластеризацией	0,00436	0,22
L-модель 3-уровневая с трихотомией	0,0056	0,36
L-модель 4-уровневая с трихотомией	0,002	0,15

Четырехуровневая модель, идентифицированная по L-алгоритму с трихотомией невязок, имеет среднюю ошибку аппроксимации в 5,4 раза меньшую, чем модель автоматизированной нейронной сети, что в процентной разности составляет 440 %. Процент улучшения аппроксимации, согласно этому же показателю, четырехуровневой модели на основе трихотомии по сравнению с трехуровневой моделью на основе кластеризации, составляет 47 %.

На рисунке 4.8 показаны коррелограммы АКФ (для лагов 1,…,15) остатков первоначальной линейной модели регрессии и четырехуровневой модели с трихотомией, упорядоченных в соответствии с возрастанием выходной величины коэффициента насыщения *КН*.



а) б)

Рисунок 4.8 – Изменение коррелограмм остатков КН от линейной регрессии (а) к четырехуровневой модели (б)

Из рисунка 4.8 видно, что автокорреляции остатков существенно снизились после идентификации иерархической модели. При этом близость АКФ остатков иерархической модели к АКФ распределения «белого шума» свидетельствует о достаточной степени извлечения информации моделью из технологических данных, делается вывод об остановке L-алгоритма на идентификации четырехуровневой модели.

4.4. Задача управления температурным режимом стадии диффузии производства сахара

Процесс получения сахара представляет собой многостадийные сложные операции, включающие следующие основные этапы: предварительная обработка свеклы; получение диффузионного сока и дальнейшая его очистка; сгущение диффузионного сока до сиропа; кристаллизация сахара (см. [21, 55, 63]).

Поддержание оптимального технологического режима, необходимого при управлении производственным процессом экстрагирования (извлечения) сахарозы из свекловичной стружки, можно добиться путем регулирования следующих технологических параметров: качество свекловичной стружки и

питательной воды, температура, откачка диффузионного сока, длительность процесса экстрагирования, инфицированность сокостружечной смеси, возврат жомопрессовой воды, содержание мезги (см. [32-33]). Наиболее важным фактором при свеклосахарном производстве из перечисленных выше является температура, в связи с чем рассмотрим возможность управления температурным режимом стадии диффузии получения сахара более подробно.

В работах [32-33] представлены результаты применения квазистатических окрестностных систем в задаче управления температурным режимом стадией диффузии производства сахара в диффузионном аппарате колонного типа. В данной работе процесс получения диффузионного сока рассматривается как система, элементами которой являются ошпариватель, диффузионная колонна и подогреватель (см. [55, 63]).

Для математической формализации технологического процесса в работах [32-33] была построена окрестностная структура (рисунок 4.9), представляющая собой орграф. Узлами орграфа являются следующие стадии процесса: узел 1 – получение сокостружечной смеси в ошпаривателе; узел 2 – технологический поток диффузионного сока из ошпаривателя; узел 3 – процесс экстракции сахара в диффузионной колонне; узел 4 – операция выгрузки сырого жома в верхней части диффузионной колонны; узлы 0 и ∞ – дополнительные формальные узлы, соответствующие внешним входам и выходам системы.

Параметр T и F (вне зависимости от индексного обозначения) характеризуют температуру и расходы различного сырья, участвующего в процессе диффузии и экстрагирования, соответственно.

Переменными состояния окрестностной структуры являются $T_{\text{сс}}$, $T_{\text{дк}}$, $T_{\text{дсо}}$, $T_{\text{жом}}$, остальные переменные являются управлениями и выходами. При этом основными параметрами, определяющими процесс экстрагирования, являются $T_{\text{сс}}$ и $T_{\text{дк}}$ (см. [63]).

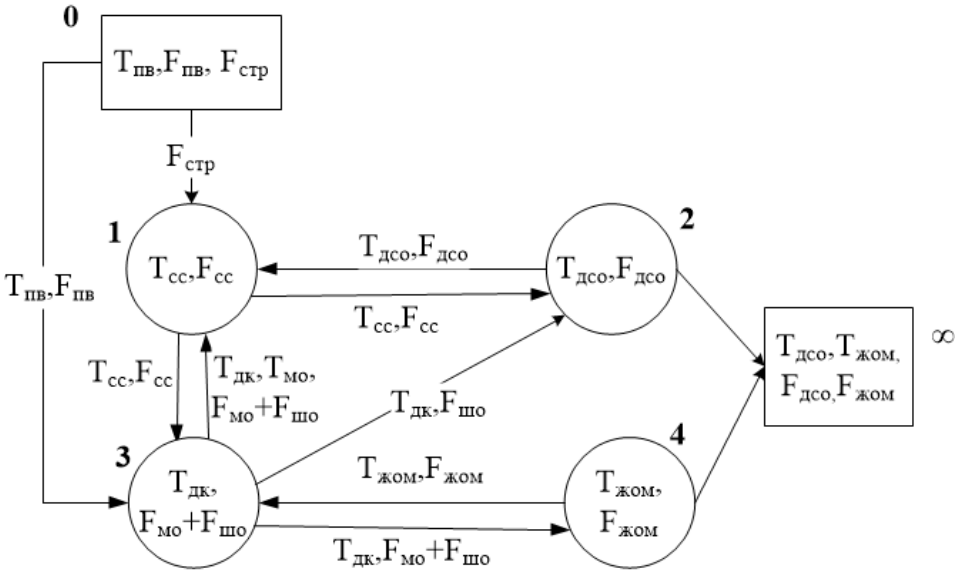


Рисунок 4.9 – Окрестностная структура процесса диффузии

В работах [32-33] выводится реляционная окрестностная система, одно из уравнений которой имеет вид:

$$T_{\text{дк}} = f(T_{\text{сс}}, T_{\text{пв}}, T_{\text{жом}}, F_{\text{шом}} + F_{\text{мо}}, F_{\text{сс}}, F_{\text{жом}}, F_{\text{пв}}), \quad (4.14)$$

являющимся важным для управления температурным режимом стадии диффузии.

Автором работ [32-33] рассматривается линейная реализация данного уравнения, условием для которой является отсутствие коррелирующих между собой пар переменных. В ходе выявленных при обработке данных автором коллинеарных пар, уравнение (4.14) приняло вид:

$$T_{\text{дк}} = f(T_{\text{сс}}, T_{\text{пв}}, F_{\text{сс}}, F_{\text{пв}}, F_{\text{жом}}), \quad (4.15)$$

или, в линейном случае:

$$T_{\text{дк}} = a_0 + a_1 T_{\text{сс}} + a_2 T_{\text{пв}} + a_3 F_{\text{сс}} + a_4 F_{\text{пв}} + a_5 F_{\text{жом}}, \quad (4.16)$$

где $T_{\text{дк}}$ – температура диффузионного сока в нижней части диффузионной колонны, $T_{\text{сс}}$ – температура сокостружечной смеси, $T_{\text{пв}}$ – температура

питательной воды, $F_{\text{сс}}$ – расход сокостружечной смеси, $F_{\text{пв}}$ – расход питательной воды, $F_{\text{жом}}$ – расход жома.

4.5. Иерархическая модель прогнозирования температурного режима стадии диффузии производства сахара

Так как идентификация уравнения (4.15) является важным этапом управления температурными режимами стадии диффузии сахара, обычную множественную линейную регрессию (4.16), идентифицируемую и используемую в работах [32-33], можно улучшить с помощью алгоритмов рекуррентной иерархической идентификации, в частности L-алгоритма.

Идентифицированная в работах [32-33] линейная регрессионная модель (4.14) (для L-алгоритма она является начальной моделью $F^1(x)$), имеет вид:

$$T_{\text{дк}} = 23{,}335 + 0{,}588T_{\text{сс}} + 0{,}079T_{\text{пв}} + 0{,}003F_{\text{сс}} + 0{,}008F_{\text{пв}} - 0{,}006F_{\text{жом}};$$

$$R = 0{,}64;\ F_{\text{расч}} = 3885{,}2;\ F_{\text{табл}} = 2{,}21;\ \bar{A} = 0{,}58\ \%. \qquad (4.17)$$

Здесь R^2 – коэффициент детерминации, $F_{\text{расч}}$ – расчетное значение критерия Фишера, $F_{\text{табл}}$ – табличное значение критерия Фишера (для p=0,05), $\bar{A}$ – средняя ошибка аппроксимации в процентах.

Применим для зависимости (4.15) L-алгоритм идентификации иерархической модели с трихотомией невязок, а также дополнительно, для сравнения, построим модели бустинга деревьев регрессии и нейронные сети по тем же исходным данным.

Результаты моделирования температуры диффузионного сока в нижней части диффузионной колонны $T_{\text{дк}}$ следующие: дерево трехуровневой модели L-алгоритма с трихотомией невязок содержит 3 узла на втором уровне (один из которых является листом) и 6 листьев на третьем уровне. На всех уровнях идентифицировались линейные модели регрессий.

Разработанный комплекс программ, как и в случае идентификации моделей прогнозирования характеристик клинкера, также используется в

задаче идентификации иерархической модели температуры диффузионного сока.

Оценки значений средней квадратичной ошибки, средней относительной ошибки аппроксимации в процентах, коэффициента множественной корреляции и расчетного значения критерия Фишера всех идентифицируемых моделей температуры диффузионного сока $T_{\text{дк}}$ представлены в таблице 4.6.

Таблица 4.6

Результаты идентификации различных моделей температуры диффузионного сока в нижней части диффузионной колонны $T_{\text{дк}}$

Модель	Среднеквадратичная ошибка (RMSE) и средняя ошибки аппроксимации ($\bar{A}$), коэффициент детерминации (R^2), расчетное значение критерия Фишера $F_{\text{расч}}$			
	RMSE	$\bar{A}$, %	R^2	$F_{\text{расч}}$
Линейная регрессия из работ [32-33]	0,63	0,58	0,64	3885,2
Бустинг деревьев	0,605	0,58	0,656	4200,8
Нейросеть	0,556	0,53	0,706	5394,6
L-модель 3-уровневая с трихотомией	0,277	0,18	0,922	28407

Как видим, все показатели (RMSE, $\bar{A}$, R^2) при идентификации иерархической модели оказались существенно улучшены по сравнению с обычной линейной моделью. Модель бустинга деревьев регрессии значимых улучшений при идентификации не показала, нейронные сети оказались немного лучше обычной линейной регрессии, но существенно хуже, чем иерархическая трехуровневая модель с трихотомией. Из таблицы видно, что показатель RMSE снизился более чем в 2 раза (примерно на 50%) для 3-уровневой модели по сравнению с линейной моделью, а относительная ошибка $\bar{A}$ снизилась более чем в 3 раза (примерно на 69 %).

Модели каждого листа дерева всех уровней при идентификации согласно L-алгоритму с трихотомией невязок представлены в формуле (4.18).

$$D = D_{\varepsilon} \cup D^{-} \cup D^{+},$$
$$D^{-} = D_{\varepsilon,-} \cup D^{-,-} \cup D^{-,+},$$
$$D^{+} = D_{\varepsilon,+} \cup D^{+,-} \cup D^{+,+}.$$

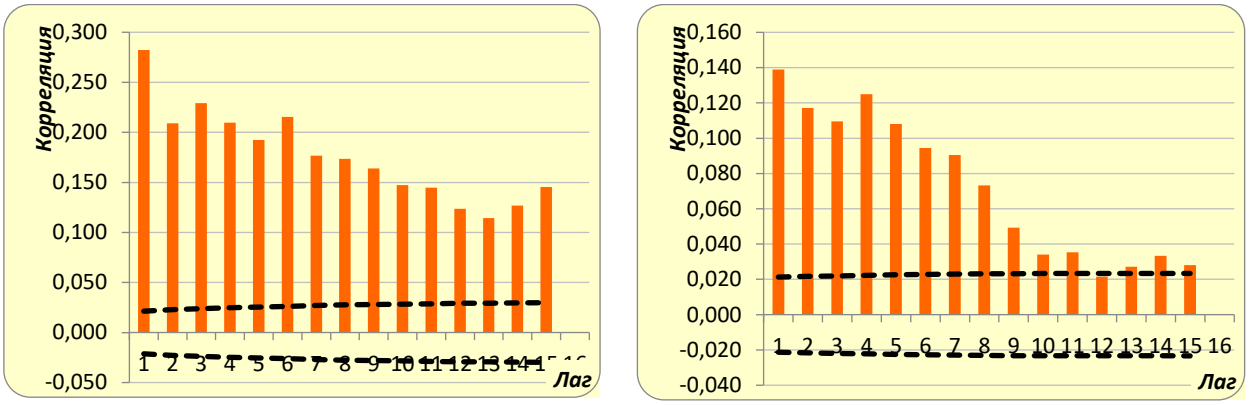
$$\begin{cases} 1-й\ уровень\ иерархии: \\ \forall d \in D_{\varepsilon},\ F^{1}(d) = 23{,}335 + 0{,}588T_{cc} + 0{,}079T_{пв} + \\ +0{,}003F_{cc} + 0{,}008F_{пв} - 0{,}006F_{жом}; \\ 2-й\ уровень\ иерархии: \\ \forall d \in D^{-},\ F^{2,-}(d) = 22{,}02 + 0{,}634T_{cc} + 0{,}043T_{пв} + \\ +0{,}004F_{cc} + 0{,}007F_{пв} - 0{,}009F_{жом}; \\ \forall d \in D^{+},\ F^{2,+}(d) = 24{,}315 + 0{,}519T_{cc} + 0{,}159T_{пв} - \\ -0{,}00026F_{cc} + 0{,}0089F_{пв} - 0{,}00744F_{жом}; \\ 3-й\ уровень\ иерархии: \\ \forall d \in D^{-,-},\ F^{3,-,-}(d) = 19{,}4115 + 0{,}673T_{cc} + 0{,}0248T_{пв} + \\ +0{,}0074F_{cc} + 0{,}007F_{пв} - 0{,}0152F_{жом}; \\ \forall d \in D^{-,+},\ F^{3,-,+}(d) = 22{,}917 + 0{,}61T_{cc} + 0{,}0595T_{пв} + \\ +0{,}0034F_{cc} + 0{,}00718F_{пв} - 0{,}00687F_{жом}; \\ \forall d \in D^{+,-},\ F^{3,+,-}(d) = 23{,}86 + 0{,}556T_{cc} + 0{,}1147T_{пв} + \\ +0{,}0017F_{cc} + 0{,}0077F_{пв} - 0{,}00645F_{жом}; \\ \forall d \in D^{+,+},\ F^{3,+,+}(d) = 24{,}7464 + 0{,}4885T_{cc} + 0{,}2355T_{пв} - \\ -0{,}009F_{cc} + 0{,}01259F_{пв} - 0{,}0079F_{жом}. \end{cases} \quad (4.18)$$

Рассмотрим коррелограммы для лагов τ = 1,…,15 остатков улучшаемой в данной работе линейной модели (4.17) и трехуровневой модели по L-алгоритму (4.18). Они представлены на рисунке 4.10, а) и б) соответственно. Следует обратить внимание на значения вертикальной шкалы корреляций.

Видно, что при переходе от линейной модели множественной регрессии к иерархической окрестностной, идентифицированной согласно L-алгоритму, наблюдается снижение величины автокорреляций невязок.

Такая ситуация означает, что иерархические модели с добавлением новых уровней точнее аппроксимируют описываемый процесс, оставляя в

остатках все меньше информации. Отсутствие какой-либо информации о процессе в остатках модели очередного уровня эквивалентно ситуации, когда АКФ для остатков мало отличима от АКФ для «белого шума».



а) б)

Рисунок 4.10 – АКФ остатков моделей (4.17) и (4.18) соответственно

Можно предположить, что анализ АКФ в иерархических моделях позволяет в некоторых случаях обойтись без тестовых данных, выборку которых используют другие алгоритмы, поскольку приближение АКФ к белому шуму можно считать индикатором остановки L-алгоритма для исключения дальнейшей переобученности. В данном случае можно остановить L-алгоритм на построенной трехуровневой модели (4.18), так как задача идентифицировать более качественную по различным показателям модель выполнена.

Итоговое дерево трехуровневой модели, идентифицированной согласно L-алгоритму с трихотомией невязок, показано на рисунке 4.11.

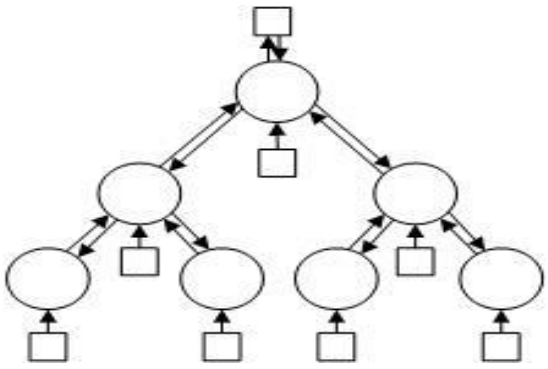


Рисунок 4.11 – Дерево 3-уровневой модели с трихотомией невязок

Можно сделать вывод о целесообразности использования L-идентификации иерархических моделей при анализе температурных режимов диффузии сахара с целью последующих этапов прогнозирования и управления процесса диффузии.

4.6. Описание комплекса проблемно-ориентированных программ

Все вычисления проведены на основе использования разработанного комплекса программ, интегрирующего алгоритмы рекуррентной идентификации иерархических окрестностных моделей, методы анализа исходных данных и контроля количества уровней иерархии (длины дерева модели) и модуль вычисления ближайшего обучающего кортежа для входных данных на этапе применения модели для решения задачи прогнозирования. Модули комплекса разработаны на основе программных средств пакетов Matlab, Mathcad и Statistica.

Процесс идентификации иерархических окрестностных моделей и их применение на основе разработанного комплекса программ можно разделить на три этапа:

1) идентификация модели, производимая или согласно R-алгоритму, или согласно L-алгоритму. Применение R-алгоритма обуславливает начальное задание структуры иерархии (дерева модели Т) в отдельном блоке модуля – количество уровней иерархии (длина дерева) N и количество узлов $n(I_s)$, подчиненных узлу I_s. Применение L-алгоритма подразумевает определение количества узлов $n(I_s)$, подчиненных узлу I_s, непосредственно в процессе идентификации, а длина дерева N заранее не задается и определяется только в момент остановки алгоритма. Описание алгоритмов представлено в параграфах 3.1-3.2. Исходными данными для обучения является обучающая выборка;

2) анализ исходных и остаточных данных, представленный в параграфе 3.3. Применение модуля анализа остаточных данных обусловлено

необходимостью контроля количества уровней иерархии дерева модели. Важным условием метода контроля количества уровней иерархии является упорядочивание остаточных данных промежуточных моделей в соответствии с возрастанием (убыванием) выходной величины (таргета). В L-алгоритме данный метод является необходимым для остановки алгоритма. В R-алгоритме метод можно применять как корректирующий заранее заданную структуру дерева T: для данной ветки дерева можно удалять все последующие вершины из дерева (провести «стрижку» дерева) или, наоборот, применить R-алгоритм идентификации повторно с другими настройками (например, увеличить количество узлов) в случае, предусмотренном данным методом анализа остаточных данных;

3) прогнозирование с помощью модели. В данном модуле осуществляется поиск ближайшего обучающего кортежа с помощью классического евклидового расстояния и определяется конкретный вид формулы общей модели, идентифицированной в виде «иерархического ряда» (см. формулу (3.2)). Поиск ближайшего обучающего кортежа является в то же время определением номера выхода (концевого узла) в структуре дерева модели T и, таким образом, обуславливает движение по дереву «снизу-вверх». При этом определяется конкретная ветка дерева и, соответственно, конкретный вид формулы общей иерархической модели.

Разработанный комплекс программ может применяться для прогнозирования различных производственных показателей и характеристик, что отражено в предыдущих параграфах данной главы применительно к задачам на цементном производстве и на производстве сахара.

4.7. Выводы по главе 4

В главе 4 получены следующие результаты:

1. Реализованы алгоритмы рекуррентной идентификации на множествах сгенерированных обучающих данных.

2. Реализован динамический алгоритм и численный метод рекуррентной идентификации (L-алгоритм) к задаче прогнозирования характеристик клинкера на цементном производстве.

3. Реализован динамический алгоритм и численный методы рекуррентной идентификации (L-алгоритм) к задаче прогнозирования температурного режима стадии диффузии производства сахара.

4. Применен метод анализа остаточных данных при работе L-алгоритма, позволяющий реализовать остановку алгоритма и определить количество необходимых для достижения соответствия между точностью и качеством прогноза уровней иерархической окрестностной модели.

5. Описан комплекс проблемно-ориентированных программ, реализующий идентификацию и применение иерархических оркестностных моделей.

ЗАКЛЮЧЕНИЕ

В работе введены иерархические окрестностные структуры и иерархические окрестностные модели. Разработаны статический и динамический алгоритмы рекуррентной идентификации иерархических окрестностных моделей. Представлена схема анализа остаточных данных промежуточных моделей в алгоритмах рекуррентной идентификации, позволяющая контролировать количество уровней иерархии. Данные алгоритмы и схема анализа остаточных данных были применены для идентификации моделей прогнозирования модульных характеристик клинкера на цементном производстве и прогнозирования температурного режима стадии диффузии производства сахара.

В рамках работы были получены следующие основные результаты:

1. Разработан и исследован новый класс окрестностных моделей, отличающийся иерархической структурой с двусторонними связями и возможностью добавления новых узлов, что позволяет расширить возможности применения окрестностных систем в задачах моделирования производственных процессов и задачах описания иерархических алгоритмов.

2. Разработан алгоритм рекуррентной идентификации иерархических окрестностных моделей на основе аппроксимации невязок промежуточных моделей, отличающийся использованием заданной иерархической кластеризации обучающих входных данных и позволяющий повысить точность моделей (R-алгоритм).

3. Разработан алгоритм и численный метод рекуррентной идентификации иерархических окрестностных моделей на основе трихотомии невязок промежуточных моделей, отличающийся построением в процессе идентификации иерархического разбиения обучающих входных данных и позволяющий улучшать точность моделей и качество прогноза (L-алгоритм).

4. Предложен метод анализа остаточных данных промежуточных моделей в алгоритмах рекуррентной идентификации, отличающийся

введением порядка на множестве обучающих данных и позволяющий контролировать количество уровней иерархии, необходимое для достижения соответствия между точностью и качеством прогноза иерархической модели.

5. Разработан комплекс проблемно-ориентированных программ, реализующих алгоритмы рекуррентной идентификации иерархических моделей, отличающихся наличием модуля вычисления ближайшего обучающего кортежа и позволяющих оценивать длину дерева модели.

СПИСОК ЛИТЕРАТУРЫ

1. Алексеев, Б.В. Технология производства цемента : учебник для сред. проф.-техн. училищ / Б.В. Алексеев. – Москва : Высшая школа, 1980. – 266 с.

2. Анищенко, В.С. Знакомство с нелинейной динамикой / В.С. Анищенко. – Москва-Ижевск : Институт компьютерных исследований, 2002. – 144 с.

3. Барский, А.Б. Нейронные сети: распознавание, управление, принятие решений / А.Б. Барский. – Москва : Финансы и статистика, 2004. – 176 с.

4. Блюмин, С.Л. Алгоритм параметрической идентификации и управления симметричными системами / С.Л. Блюмин, А.М. Шмырин, Д.А. Шмырин // Тезисы докладов Воронежской весенней математической школы «Современные методы в теории краевых задач (Понтрягинские чтения-IX)». – Воронеж : ВГУ, 1998. – 28 с.

5. Блюмин, С.Л. Билинейные окрестностные системы : монография / С.Л. Блюмин, А.М. Шмырин, О.А. Шмырина. – Липецк : ЛГТУ, 2006. – 130 с.

6. Блюмин, С.Л. Идентификация и управление окрестностными системами / С.Л. Блюмин, А.М. Шмырин, О.А. Шмырина // Идентификация систем и задачи управления: материалы международной конференции SICRO-05. – Москва : ИПУ, 2005. – С. 343-351.

7. Блюмин, С.Л. Идентификация и управление симметричными и смешанными экологическими системами / С.Л. Блюмин, А.М. Шмырин, Д.А. Шмырин // Проблемы безопасности транспортного пространства. – Липецк : ЛГТУ, 1998. – 11 с.

8. Блюмин, С.Л. Сети Петри как окрестностные системы со специальными ограничениями / С.Л. Блюмин, А.М. Шмырин, И.А. Седых // Идентификация систем и задачи управления : труды VIII Международной конференции SICPRO-09, Москва, 26-30 января 2009 г. – Москва, 2009. – С. 1550-1558.

9. Блюмин, С.Л. Новое направление в моделировании систем: окрестностные модели / С.Л. Блюмин, А.М. Шмырин, Д.А. Шмырин // Программное обеспечение автоматизированных систем управления : Международная научно-техническая конференция, 15-17 мая 2000 г., Липецк. – Липецк, 2000. – С. 15-19.

10. Блюмин, С.Л. О симметричных нечетких окрестностных матричных системах / С.Л. Блюмин, А.М. Шмырин, О.А. Шмырина // Вестник Тамбовского университета, серия «Естественные и технические науки». – 2003. – Т. 8, № 3. – С. 348-349.

11. Блюмин, С.Л. Окрестностное моделирование организационно-технических систем : монография / С.Л. Блюмин, А.М. Шмырин, И.А. Седых, С.С. Роенко, А.П. Щербаков. – Липецк : ЛЭГИ, 2013. – 104 с.

12. Блюмин, С.Л. Окрестностное моделирование сетей Петри / С.Л. Блюмин, А.М. Шмырин, И.А. Седых, В.Ю. Филоненко. – Липецк : ЛЭГИ, 2010. – 124 с.

13. Блюмин, С.Л. Окрестностные системы / С.Л. Блюмин, А.М. Шмырин. – Липецк : ЛЭГИ, 2005. – 132 с.

14. Блюмин, С.Л. Оптимальное управление смешанными окрестностными системами / С.Л. Блюмин, А.М. Шмырин // Современные методы теории функций и смежные проблемы : тезисы докладов Воронежской зимней математической школы. – Воронеж, 1999. – С. 42.

15. Блюмин, С.Л. От систем на графах к окрестностным системам / С.Л. Блюмин, А.М. Шмырин // Математическое моделирование систем. Методы, приложения и средства : труды конференции. – Воронеж, 1998. – С. 33-41.

16. Блюмин, С.Л. Представления нелинейных нечетко-окрестностных систем / С.Л. Блюмин, А.М. Шмырин, О.А. Шмырина // Проблемы управления. – 2005. – № 2. – С. 37-40.

17. Блюмин, С.Л. Симметричные, смешанные и билинейные окрестностные модели / С.Л. Блюмин, О.А. Шмырина // Экономика и

управление, математика : сборник научных трудов. – Липецк, 2002. – С. 44-48.

18. Блюмин, С.Л. Моделирование нечётких сетей Петри окрестностными системами / С.Л. Блюмин, А.М. Шмырин, И.А. Седых // Нечеткие системы и мягкие вычисления (НСМВ-2008) : сборник научных трудов Второй Всероссийской научной конференции с международным участием. – Ульяновск, 2008. – С. 96-104.

19. Бодянский, Е.В. Искусственные нейронные сети: архитектуры, обучение, применения / Е.В. Бодянский, О.Г. Руденко. – Харьков : ТЕЛЕТЕХ, 2004. – 369 с.

20. Болтянский, В.Г. Математические методы оптимального управления / В.Г. Болтянский. – Москва : Наука, 1969. – 408 с.

21. Бугаенко, И.Ф. Основы сахарного производства / И.Ф. Бугаенко. – Москва : Международная сахарная компания, 2002. – 332 с.

22. Воронин, А.А. Математические модели организаций : учебное пособие / А.А. Воронин, М.В. Губко, С.П. Мишин, Д.А. Новиков. – Москва : ЛЕНАНД, 2008. – 360 с.

23. Воронин, А.А. Оптимальные иерархические структуры / А.А. Воронин, С.П. Мишин. – Москва : ИПУ РАН, 2003. – 214 с.

24. Голованова, Л.В. Общая технология цемента : учебник для средних профессионально-технических училищ. – Москва : Стройиздат, 1984. – 118 с.

25. Губко, М.В. Математические модели оптимизации иерархических структур / М.В. Губко. – Москва : ЛЕНАНД, 2006. – 264 с.

26. Данилов, Ю.А. Лекции по нелинейной динамике. Элементарное введение / Ю.А. Данилов. – Изд. 2-е, испр. – Москва : КомКнига, 2006. – 208 с.

27. Думачев, В.Н. Эволюция антагонистически-взаимодействующих популяций на базе двумерной модели Ферхюльста-Пирла / В.Н. Думачев, В.А. Родин // Математическое моделирование. – Т. 17, № 7. – 2005. – С. 11-22.

28. Дыхта, В.А. Динамические системы в экономике. Введение в анализ одномерных моделей : учебное пособие / В.А. Дыхта. – Иркутск : Изд-во БГУЭП, 2003. – 178 с.

29. Дюран, Б. Кластерный анализ / Б. Дюран. – Москва : Статистика, 1977. – 128 с.

30. Занг, В.Б. Синергетическая экономика. Время и перемены в нелинейной экономической теории / В.Б. Занг ; пер. с англ. – Москва : Мир, 1999. – 335 с.

31. Калман, Р. Очерки по математической теории систем / Р. Калман, П. Фалб, М. Арбиб. – Москва : Мир, 1971. – 400 с.

32. Канюгина, А.С. О задаче управления температурным режимом стадии диффузии производства сахара / А.С. Канюгина // Вестник Воронежского государственного технического университета. – 2019. – Т. 15, № 2. – С. 51-63.

33. Канюгина, А.С. Разработка квазистатических окрестностных систем и их применение в задаче управления температурным режимом стадии диффузии производства сахара : специальность 05.13.01 «Системный анализ, управление и обработка информации» : диссертация на соискание ученой степени кандидата технических наук / Канюгина Анастасия Сергеевна. – Липецк : ЛГТУ, 2019. – 126 с.

34. Карабутов, Н.Н. Информационные аспекты идентификации окрестностных и нечетко-окрестностных систем / Н.Н. Карабутов, А.М. Шмырин // Идентификация систем и задачи управления : труды V Международной конференции SICPRO-06. – Москва : ИПУ, 2006. – С. 244-254.

35. Карабутов, Н.Н. Окрестностные и нечетко-окрестностные модели пространственно-распределенных систем / Н.Н. Карабутов, А.М. Шмырин, О.А. Шмырина // Приборы и системы. Управление, контроль, диагностика. – 2005. – № 12. – С. 19-22.

36. Карабутов, Н.Н. Окрестностные системы: идентификация и оценка состояния / Н.Н. Карабутов, А.М. Шмырин. – Липецк : ЛЭГИ, 2005. – 132 с.

37. Корниенко, Н.А. Моделирование неполносвязных нейронных сетей окрестностными системами / Н.А. Корниенко, А.М. Шмырин, И.А. Седых, Т.А. Шмырина // Материалы Четвёртой международной конференции «Управление развитием крупномасштабных систем (MLSD′2010)». Т. 1. – Москва, 2010. – С. 325-327.

38. Кристофидес, Н. Теория графов. Алгоритмический подход / Н. Кристофидес. – Москва : Мир, 1978. – 432 с.

39. Круглов, В.В. Искусственные нейронные сети. Теория и практика / В.В. Круглов, В.В. Борисов. – Москва : Горячая линия – Телеком, 2002. – 382 с.

40. Кузнецов, А.П. Введение в физику нелинейных отображений / А.П. Кузнецов, А.В. Савин, Л.В. Тюрюкина. – Саратов : Научная книга, 2010. – 134 с.

41. Кузнецов, С.П. Динамический хаос / С.П. Кузнецов. – Москва : Физматлит, 2001. – 296 с.

42. Лескин, А.А. Сети Петри в моделировании и управлении / А.А. Лескин, П.А. Мальцев, А.М. Спиридонов. – Ленинград : Наука, 1989. – 133 с.

43. Лимановская, О.В. Основы машинного обучения : учебное пособие / О.В. Лимановская, Т.И. Алферьева ; Министерство науки и высшего образования РФ. – Екатеринбург : Изд-во Уральского университета, 2020. – 88 с.

44. Малинецкий, Г.Г. Современные проблемы нелинейной динамики / Г.Г. Малинецкий, А.Б. Потапов. – Москва : Эдиториал УРСС, 2000. – 336 с.

45. Мишачев, Н.М. Две схемы иерархической идентификации квазилинейных моделей / Н.М. Мишачев, А.М. Шмырин, А.П. Щербаков // Вестник Воронежского государственного технического университета. – 2023. – Т. 19, № 1. – С. 7-13.

46. Мишачев, Н.М. Дискретные системы и окрестностные структуры / Н.М. Мишачев, А.М. Шмырин // Вестник Тамбовского университета. Серия: естественные и технические науки. – 2018, Т. 23. № 123. – С. 473-478.

47. Мишачев, Н.М. Метаструктурная идентификация : монография / Н.М. Мишачев, А.М. Шмырин. – Воронеж : ООО «РИТМ», 2019. – 189 с.

48. Мишачев, Н.М. Окрестностные структуры и метаструктурная идентификация / Н.М. Мишачев, А.М. Шмырин // Таврический вестник информатики и математики. – 2017. – Т. 37, № 4 (37). – С. 87-95.

49. Нильсен, Э. Практический анализ временных рядов: прогнозирование со статистикой и машинное обучение. / Э. Нильсен ; пер. с англ. – Санкт-Петербург : ООО «Диалектика», 2021. – 544 с.

50. Норри, Д. Введение в метод конечных элементов / Д. Норри, Ж. де Фриз. – Москва : Мир, 1981. – 155 с.

51. Олдендерфер, М.С. Факторный, дискриминантный и кластерный анализ / М.С. Олдендерфер, Р.К. Блэшфилд ; пер. с англ. ; под. ред. И.С. Енюкова. – Москва : Финансы и статистика, 1989. – 215 с.

52. Петерс, Э. Фрактальный анализ финансовых рынков: применение теории хаоса в инвестициях и экономике / Э. Петерс. – Москва : Интернет-трейдинг, 2004. – 304 с.

53. Петерс, Э. Хаос и порядок на рынках капитала. Новый аналитический взгляд на циклы, цены и изменчивость рынка / Э. Петерс ; пер. с англ. – Москва : Мир, 2000. – 333 с.

54. Подвальный, С.Л. Иерархическая идентификация параметров нелинейных динамических систем / С.Л. Подвальный, Е.М. Васильев // Сборник трудов XIII Всероссийского совещания по проблемам управления ВСПУ-2019, г. Москва, 17–20 июня 2019 года / Институт проблем управления им. В.А. Трапезникова РАН. – Москва, 2019. – С. 517-521.

55. Сапронов, А.Р. Технология сахара : учебник / А.Р. Сапронов, Л.А. Сапронова, С.В. Ермолаев. – Санкт-Петербург : Профессия, 2013. – 296 с. – ISBN 978-5-904757-45-8.

56. Седых, И.А. Двухуровневые полиномиальные динамические окрестностные модели с переменными окрестностями и их параметрическая идентификация / И. А. Седых // Вести высших учебных заведений Черноземья. – 2018. – № 1 (51). – С. 57–65.

57. Седых, И.А. Динамические окрестностные сети / И.А. Седых // Вестник Уфимского государственного авиационного технического университета. – 2018. – Т. 22, № 3 (81). – С. 124–130.

58. Седых, И.А. Идентификация линейных динамических окрестностных моделей с нечеткой иерархической структурой / И.А. Седых // Вестник Воронежского государственного технического университета. – 2019. – Т. 15, № 4. – С. 7-13.

59. Седых, И.А. Окрестностное моделирование уровня подземных вод месторождения цементных известняков и глин / И.А. Седых // Актуальные направления научных исследований XXI века: теория и практика. – 2018. – Т. 6, № 6 (42). – С. 314-316.

60. Седых, И.А. Прогнозирование уровня подземных вод месторождения цементного сырья на основе динамических окрестностных моделей / И.А. Седых // Вестник Донского государственного технического университета. – 2018. – Т. 18, № 3. – С. 326-332.

61. Свидетельство о регистрации программы для ЭВМ RU 2015617475 Российская Федерация Исследование и решение систем методом ортонормализации Грама-Шмидта : № 2015612504 : заявл. 01.04.2015 : опубл. 10.07.2015 / В.В. Семёнова, А.М. Шмырин, И.А. Седых, А.П. Щербаков, А.Г. Ярцев : Правообл. Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Липецкий государственный технический университет» (ФГБОУ ВПО ЛГТУ) – Версия операционной системы: Windows XP и выше. – Объем: 88,5 Кб.

62. Сёмина, В.В. Энергосберегающая система управления производственной вентиляцией и фильтрацией на основе слабосвязных окрестностных систем / В.В. Сёмина, Н.М. Мишачев, А.М. Шмырин // Энерго-

и ресурсосбережение – XXI век : материалы XIX Международной научно-практической конференции. – Орел, 2021. – С. 129-134.

63. Силин, П.М. Технология сахара : учебник / П.М. Силин. – 2-е изд., перераб. И доп. – Москва : Пищевая промышленность, 1967. – 632 с.

64. Солонина, А.И. Цифровая обработка сигналов в зеркале MATLAB : учебное пособие / А.И. Солонина. – Санкт-Петербург : БХВ-Петербург, 2018. – 560 с. – ISBN 978-5-97-753946-3.

65. Томилин, А.А. Особенности аппарата формирования организационных структур на основе окрестностно-временных моделей / А.А. Томилин // III Всероссийская молодежная конференция по проблемам управления (ВМКПУ'2008) : труды научно-практической конференции / под ред. Д.А. Новикова, З.К. Авдеевой. – Москва, 2008. – С. 209-210.

66. Хайкин, С. Нейронные сети : полный курс / С. Хайкин. – Москва : Издательский дом «Вильямс», 2006. – 1104 с.

67. Хакен, Г. Синергетика / Г. Хакен. – Москва : Мир, 1980. – 404 с.

68. Хасти, Т. Основы статистического обучения: интеллектуальный анализ данных, логический вывод и прогнозирование / Т. Хасти, Р. Тибширани, Дж. Фридман. – 2-е изд. ; пер. с англ. – Санкт-Петербург : ООО «Диалектика», 2020. – 768 с. – ISBN 978-5-907144-42-2.

69. Царьков, В.А. Динамические модели в экономике. Теория и практика экономической динамики / В.А. Царьков. – Москва : Экономика, 2007. – 217 с. – ISBN 978-5-282-02695-5.

70. Чуличков, А.И. Математические модели нелинейной динамики / А.И. Чуличков. – 2-е изд., испр. – Москва : ФИЗМАТЛИТ, 2003. – 296 с.

71. Ширяев, В.И. Финансовые рынки, нейронные сети, хаос и нелинейная динамика: учебное пособие / В.И. Ширяев. – 2-е изд., испр. и доп. – Москва : Книжный дом «ЛИБРОКОМ», 2009. – 232 с.

72. Шмырин, А.М. Агрегирование окрестностных систем в модели вентиляции цеха цементного / А.М. Шмырин, Н.М. Мишачев, В.В. Семина //

Вестник Тамбовского университета, серия «Естественные и технические науки». – 2017. Т. 22. № 6. – С. 1346-1354.

73. Шмырин, А.М. Алгоритмы выбора окрестностей нейронов [Текст] / А.М. Шмырин, Н.А. Корниенко, С.С. Роенко, О.А. Митина // Материалы IV Международной научной конференции «Современные проблемы прикладной математики, теории управления и математического моделирования)» (ПМТУММ-2011). – Воронеж: ВГУ, 2011. – С. 312-313.

74. Шмырин, А.М. Алгоритмизация процессов смешанного управления пространственно-распределенными системами на основе нечетко-окрестностных моделей: дис. д-ра техн. наук: 05.13.18 / А.М. Шмырин. – Воронеж, 2007. – 428 с.

75. Шмырин, А.М. Алгоритмы идентификации и управления функционированием окрестностных систем, полученных на основе сетей Петри / А.М. Шмырин, И.А. Седых // Управление большими системами. – Москва : ИПУ РАН, 2009. – № 24. – С. 18-33.

76. Шмырин, А.М. Алгоритмы идентификации и управления функционированием окрестностными системами, полученными на основе сетей Петри / А.М. Шмырин, И.А. Седых // 5 Всероссийская школа-семинар молодых учёных «Управление большими системами». Т.1. – Липецк, 2008. – С. 72-77.

77. Шмырин, А.М. Виды нечёткости в динамических окрестностных моделях / А.М. Шмырин, И.А. Седых // Вестник Тамбовского университета. – 2012. – Т. 17, № 4. – С. 1129-1130.

78. Шмырин, А.М. Два подхода к исследованию общего параметрического уравнения окрестностной модели печи обжига клинкера / А.М. Шмырин, И.А. Седых, А.П. Щербаков, А.Г. Ярцев // Системы управления и информационные технологии. – 2015. – № 1.1 (59). – С. 185-189.

79. Шмырин, А.М. Дискретные модели в классе окрестностных систем / А.М. Шмырин, И.А. Седых // Вестник Тамбовского университета. – 2012. – Т. 17, № 3. – С. 867-871.

80. Шмырин, А.М. Допустимое смешанное управление билинейными окрестностными системами / А.М. Шмырин, О.А. Шмырина // Современные проблемы информатизации в непромышленной сфере и экономике : сборник трудов (по итогам X Международной открытой научной конференции). – Воронеж, 2005. – № 4. – С. 129-130.

81. Шмырин, А.М. Задача достижимости с частично заданными параметрами для динамических недетерминированных окрестностных моделей нейронных сетей Петри / А.М. Шмырин, И.А. Седых // Управление большими системами : материалы IX Всероссийской школы-конференции молодых учёных. Т. 1. – Тамбов : Липецк, 2012. – С. 108-110.

82. Шмырин, А.М. Идентификация линейных окрестностных систем, представляющих сети Петри / А.М. Шмырин, И.А. Седых // Перспективы развития информационных технологий : сборник материалов 1 Всероссийской научно-практической конференции. – Новосибирск, 2008. – С. 91-97.

83. Шмырина, О.А. Информационные аспекты идентификации билинейных окрестностных систем / О.А. Шмырина // Вестник ЛГТУ. – 2005. – № 1 (13). – С. 33-36.

84. Исследование окрестностной модели печи обжига клинкера с учетом допустимых значений параметров / А.М. Шмырин, И.А. Седых, А.П. Щербаков, А.Г. Ярцев // Вестник Липецкого государственного технического университета. – 2015. – № 2 (24). – С. 11-14.

85. Шмырин, А.М. Исследование поведения окрестностной системы с учетом допустимых значений параметров / А.М. Шмырин, А.П. Щербаков, А.Г. Ярцев // Вестник Тамбовского университета, серия «Естественные и технические науки». – 2013. – Т. 18, № 5-2. – С. 2747-2748.

86. Шмырин, А.М. Классификация билинейных окрестностных моделей / А.М. Шмырин, И.А. Седых // Вестник Тамбовского университета. – 2012. – Т. 17, № 5. – С.1366-1370.

87. Шмырин, А.М. Классы окрестностных систем, полученных на основе сетей Петри / А.М. Шмырин, И.А. Седых // Вестник Тамбовского университета. – 2009. – Т. 14, № 4. – С. 840-841.

88. Шмырин, А.М. Методы нелинейного анализа при исследовании характеристик производства клинкера / А.М. Шмырин, И.А. Седых, А.П. Щербаков // Вестник Тамбовского университета, серия «Естественные и технические науки». – 2014. – Т. 19. № 3. – С. 923-926.

89. Моделирование иерархических окрестностных систем / А.М. Шмырин, И.А. Седых, Н.А. Корниенко [и др.] // Материалы конференции «Управление в технических системах» (УТС-2010). – Санкт-Петербург, 2010. – С.43-46.

90. Шмырин, А.М. Наличие экстремумов параметрического уравнения печи обжига клинкера / А.М. Шмырин, И.А. Седых, А.П. Щербаков, А.Г. Ярцев // Вести высших учебных заведений Черноземья. – 2015. – № 1(39). – С. 62-67.

91. Шмырин, А.М. Некоторые вопросы окрестностной нечёткости модели производства полиола / А.М. Шмырин, А.Г. Ярцев // Современные методы прикладной математики, теории управления и компьютерных технологий : сборник трудов XI Международной конференции «ПМТУКТ-2018». – Воронеж, 2018. – С. 309-312.

92. Шмырин, А.М. Нечетко-окрестностные нелинейные системы в координатной форме / А.М. Шмырин // Современные проблемы математики, механики, информатики : тезисы докладов Международной конференции. – Тула, 2003. – С. 346-347.

93. Шмырин, А.М. Нечетко-окрестностные системы / А.М. Шмырин // Проблемы непрерывного образования: проектирование, управление, функционирование : материалы международной научно-методической конференции. – Липецк, 2003. – С. 69-72.

94. Шмырин, А.М. Обобщение дискретных моделей окрестностными системами / А.М. Шмырин, И.А. Седых, Н.А. Корниенко, Т.А. Шмырина //

Материалы конференции с международным участием «Технические и программные средства систем управления, контроля и измерения» (УКИ 10). – Москва, 2010. – С. 207-208.

95. Шмырин, А.М. Общие билинейные дискретные модели / А.М. Шмырин, И.А. Седых, А.П. Щербаков // Вестник Воронежского государственного технического университета. – 2014. – Т. 10, № 3-1. – С. 44-49.

96. Шмырин, А.М. Окрестностные иерархические системы с переменными окрестностями / А.М. Шмырин, И.А. Седых, Н.А. Корниенко, Н.А. Сапожников, О.А. Митина, Т.А. Шмырина // Вестник Тамбовского Университета, серия «Естественные и технические науки», – 2010. – Т. 15, № 6. – С. 1941-1942.

97. Окрестностное моделирование двумерных нелинейных динамических систем / А.М. Шмырин, И.А. Седых, А.П. Щербаков [и другие] // Вестник Тамбовского университета. – Т. 18, вып. 1. – Тамбов, 2013. – С. 81-88.

98. Шмырин, А.М. Окрестностное моделирование распределённых нелинейных динамических систем / А.М. Шмырин, С.С. Роенко, А.П. Щербаков // Управление большими системами: материалы IX Всероссийской школы-конференции молодых учёных. Т. 1. – Тамбов : Липецк, 2012. – С. 104-107.

99. Шмырин, А.М. Окрестностные системы с динамическими окрестностями / А.М. Шмырин, И.А. Седых, Н.А. Корниенко, Т.А. Шмырина // Параллельные вычисления и задачи управления : труды 5 Международной конференции PACO-10. – Москва, 2010. – С. 4-8.

100. Шмырин, А.М. Окрестностный подход к моделированию распределённых динамических систем / А.М. Шмырин, В.В. Кавыгин, С.С. Роенко, А.П. Щербаков // Фундаментальные и прикладные проблемы модернизации современного машиностроения и металлургии: сборник научных трудов международной научно-технической конференции,

посвященной 50-летию кафедры технологии машиностроения ЛГТУ. В 2 ч. Ч. 2 / под общ. ред. проф. А.М. Козлова. – Липецк, 2012. – С. 321-326.

101. Шмырин, А.М. Определение предельных точек для модели системы теплоснабжения / А.М. Шмырин, А.П. Щербаков, А.Г. Ярцев // Современные тенденции в образовании и науке : материалы Международной научно-практической конференции. В 26 ч. Т. 9. – Тамбов, 2013. – С. 155-158.

102. Шмырин, А.М. Параметрическая билинейная окрестностная модель системы теплоснабжения / А.М. Шмырин, А.П. Щербаков, А.Г. Ярцев // Современные тенденции в образовании и науке : материалы Международной научно-практической конференции. В 26 ч. Т. 9. – Тамбов, 2013. – С. 155-158.

103. Шмырин, А.М. Параметрическая идентификация окрестностной модели процесса воздухообмена в производственном помещении / А.М. Шмырин, В.В. Семина, Е.П. Трофимов // Вестник Тамбовского университета, серия «Естественные и технические науки». – 2017. – Т. 22, № 3. – С. 605-610.

104. Параметрическое окрестностное моделирование печи обжига клинкера / А.М. Шмырин, И.А. Седых, А.П. Щербаков [и другие] // Вестник тамбовского университета, серия: «Естественные и технические науки». – 2014. – Т. 19, № 3. – С. 927-930.

105. Свидетельство о регистрации программы для ЭВМ RU 2015614243 Российская Федерация. Расчет корреляционных размерностей и доли случайного хаоса по данным временного ряда : № 2015610763 : зарег. 13.02.2015 : опубл 10.04.2015 / А.М. Шмырин, А.П. Щербаков, И.А. Седых, А.Г. Ярцев : правообл. Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Липецкий государственный технический университет» (ФГБОУ ВПО ЛГТУ). – Версия операционной системы: Windows XP/Vista/7/8/10. – Объем: 0,6 Мб.

106. Шмырин, А.М. Реляционные окрестностные системы / А.М. Шмырин, Н.М. Мишачев, А.С. Канюгина // Современные сложные системы

управления: HTCS'2017 : материалы XII Международной научно-практической конференции. – Липецк, 2017. – С. 74-77.

107. Шмырин, А.М. Смешанное управление окрестностными системами / А.М. Шмырин // Системы управления и информационные технологии. – 2007. – № 1 (27). – С. 26-30.

108. Шмырин, А.М. Смешанное управление в чётких и нечётких окрестностных моделях / А.М. Шмырин, И.А. Седых // VIII Международная конференция по финансово-актуарной математике и смежным вопросам ФАМ`2009. – Красноярск, 2009. – С. 124-129.

109. Шмырин, А.М. Смешанное управление нечёткими окрестностными системами, полученными на основе нечётких сетей Петри / А.М. Шмырин, И.А. Седых, Л.Д. Арестова // Информационные технологии моделирования и управления. – 2009. – № 1. – С. 92-97.

110. Структурное окрестностное моделирование систем промышленной вентиляции / А.М. Шмырин, В.В. Сёмина, О.А. Мещерякова, Е.А. Лукьянова // Таврический вестник информатики и математики. – 2017. – № 4. – С. 96-105.

111. Щербаков, А.П. Генерирование тестовых данных для регрессионной идентификации квазилинейных иерархических моделей / А.П. Щербаков // Вестник Липецкого государственного технического университета. – 2022. – № 3 (49). – С. 41-47.

112. Щербаков, А.П. Иерархическая квазилинейная модель прогнозирования свойств клинкера / А.П. Щербаков // Вестник Воронежского государственного технического университета. – 2023. – Т. 19, № 3. – С. 17-22.

113. Щербаков, А.П. Идентификация мультимодальных окрестностных систем / А.П. Щербаков, А.М. Шмырин, Н.М. Мишачев // Системы управления и информационные технологии. – 2022. –№ 3 (89). – С. 24-28.

114. Щербаков, А.П. К вопросу исследования хаотичности входного процесса / А.П. Щербаков. – Елец, 2014. – С. 234-237.

115. Щербаков, А.П. Моделирование логистического отображения билинейными окрестностными системами / А.П. Щербаков, А.М. Шмырин // Современные проблемы информатизации в экономике и обеспечении безопасности : сборник трудов. – Воронеж, 2012. – № 17. – С. 119-123.

116. Щербаков, А.П. Представление нейронных сетей в виде общих окрестностных моделей / А.П. Щербаков // Актуальные проблемы естественных наук и их преподавания : материалы научной конференции. – Липецк, 2016. – С. 282-288.

117. Щербаков, А.П. Трехуровневая иерархическая регрессионная модель / А.П. Щербаков // Нано-био-технологии. Теплоэнергетика. Математическое моделирование : материалы Международной научно-практической конференции, г. Липецк, 27-27 февраля 2023 г. – Липецк, 2023. – С. 283-288.

118. Щербаков, А.П. L-алгоритм моделирования переменных состояния стадии диффузии производства сахара / А.П. Щербаков // Вестник Липецкого государственного технического университета. – 2023. – № 3 (52). – С. 21-27.

119. Яновский, Л.П. Анализ состояния финансовых рынков на основе методов нелинейной динамики / Л.П. Яновский, Д.А. Филатов // Экономический анализ: теория и практика. – 2005. – № 17 (50). – С. 5-15.

120. Свидетельство о государственной регистрации программы для ЭВМ RU 2015610766. Российская Федерация. Получение общего параметрического уравнения окрестностной модели : № 2014661696 : зарег. 19.11.2014 : опубл. 16.01.2015 / А.Г. Ярцев, А.М. Шмырин, И.А. Седых, А.П. Щербаков ; правообл. Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Липецкий государственный технический университет» Версия операционной системы: Windows XP. – Объем: 1,1 Мб.

121. Basu, A. Metagraphs and their applications / A. Basu, R. Blanning // Springer, 2007. – 174 p.

122. Breiman, L. Classification and Regression Trees. / L. Breiman, J. Friedman, R. Olshen, C. Stone // Wadsworth, Belmont, CA, – 1984.

123. Etipola, S High fidelity clinker quality forecasting and controlling system using machine learning / S. Etipola, C. Dassanayaka, S. Samarasinghe, R. Mallawarachchi // 2020 international conference on civil, architectural and environmental engineering, Christchurch, New Zealand on November 23-25, 2020. – 2021. – Vol. 2409. – P. 8.

124. Hurst H.E. Long-term Storage of Reservoirs / H.E. Hurst // Transactions of the American Society of Civil Engineers. – 1951. – 156 p.

125. Mishachev, N.M. Approximation of curves in phase space by solutions of control system / N.M. Mishachev, A.M. Shmyrin, A.P. Shcherbakov // 2022 4rd International Conference on Control Systems, Mathematical Modeling, Automation and Energy Efficiency, SUMMA 2022. – New Jersey, 2022. – P. 46-48.

126. Mishachev, N.M. Polymodal neighborhood systems in system modeling course / N.M. Mishachev, A.M. Shmyrin, A.P. Shcherbakov // 2022 2nd International Conference on Technology Enhanced Learning in Higher Education, TELE 2022. – New Jersey, 2022. – p. 152-155.

127. Nilsson, N.J. Introduction to machine learning / N.J. Nilsson // Robotics Laboratory Department of Computer Science, Stanford University, Stanford, 1998. – 179 p.

128. Shmyrin, A.M. Application of mixed control for determining the heat transfer coefficient of a heat exchanger / A.M. Shmyrin A.M., A.G. Yartsev // International Journal of Applied Engineering Research. – 2017. – Vol. 12, № 20. – P. 10339-10341.

129. Shmyrin, A.M. The classification of the dinamic nondetermination neighborhood’s models / A. Shmyrin, I. Sedykh // Interactive Systems and Technologies: the Problems of Human-Computer Interaction. Vol. 3. – Collection of scientific papers. – Ulyanovsk, 2009. – P. 442-444.

130. Research trilinear neighborhood model of clinker kiln / A.M. Shmyrin, Sedykh I.A., Shcherbakov A.P., Yartsev A.G. // Modern informatization problems

in simulation and social technologies Proceedings of the XX-th International Open Science Conference. – 2015. – P. 202-206.

131. Takagi T. Fuzzy identification of systems and its applications to modeling and control / T. Takagi, M. Sugeno // IEEE transactions on systems, man, and cybernetics. – 1985. – Vol. 15, № 1. – P. 116–132.

132. Quinlan, J.R. (1986). Induction of decision trees / J.R. Quinlan // Machine Learning. – 1986. – № 1(1). – P. 81-106.

133. Quinlan, J. R. C4.5: Programs for Machine learning / J.R. Quinlan // Morgan Kaufmann Publishers, 1993.

134. Stability analysis and Nonlinear Observer Design Using Takagi-Sugeno fuzzy Models / Z. Lendek, Th.-M. Guerra, R. Babuska, B. De Schutter // Springer. – 2011 – № 1. – 196 p. – DOI 10.1007/978-3-642-16776-8.

ПРИЛОЖЕНИЯ

Приложение 1

Справки об использовании результатов диссертационной работы

АО «Липецкцемент»
ул. Ковалева, владение 126 Б, этаж 2, офис 5, г. Липецк
Россия, 398007
тел.: +7 (4742) 48 18 08, факс +7 (4742) 48 18 01
8-800-700-63-63 | e-mail: lipcem@eurocem.ru
www.eurocement.ru

10.08.2020 № 1/ЛЦ-794/20
на №_____от_____

Справка
об использовании результатов диссертационной работы
Щербакова Артема Петровича

Настоящая справка составлена в том, что результаты диссертационной работы Щербакова А.П., посвящённой разработке алгоритмов анализа временных рядов характеристик производства клинкера, алгоритмов идентификации окрестностной модели с применением методов фрактального анализа и смешанного управления производства цемента, рассмотрены применительно к задачам математического моделирования, технологического анализа, а также управления технико-экономическими показателями на примере производства цемента АО «Липецкцемент».

В частности:

1) выполнен анализ возможности использования результатов работы Щербакова А.П. при планировании, анализе и управлении техническими и технологическими показателями производства цемента заводом АО «Липецкцемент»;

2) на основе выполненного анализа производственных данных завода АО «Липецкцемент» установлено, что предлагаемые методы работоспособны, могут быть использованы при управлении крупными производственными подразделениями (в данном случае, печами обжига клинкера) и позволяют обеспечивать повышение эффективности их работы;

3) разработанные математические модели и методы управления рекомендуются к использованию при управлении крупными производственными подразделениями предприятий.

Генеральный директор **И.М. Звягин**

Российская Федерация Липецкая область Акционерное общество «Липецкцемент» ОТДЕЛ ДЕЛОПРОИЗВОДСТВА г. Липецк

АО «Агропромышленное объединение «Аврора»

398002 г. Липецк, улица Тельмана, дом 11 Тел/Факс (4742)34-59-62, (47471)5-22-09
ИНН 4825003761 КПП 482501001 Задонское ОСБ 3827 г. Задонск р/с 40702810935060100255
Липецкое ОСБ 8593 г. Липецк к/с 30101810800000000604 БИК 044206604 ОКПО 00897763
ОКОНХ 21110,21120,21130,80100,71100
E-mail: avrora@lipetsk.ru
Генеральный директор Уваркин Александр Сергеевич
Подразделение «Хмелинецкий сахарный завод»
Тел./факс (47471)3-55-30, 3-54-63
hmelinec@apo-avrora.ru

Справка
об использовании результатов диссертационной работы
Щербакова Артема Петровича

Настоящей справкой удостоверяется, что результаты диссертационной работы Щербакова А.П., посвящённой разработке методов иерахической идентификации квазилинейных окрестностных моделей, рассмотрены применительно к задачам математического моделирования и совершенствования управления технологическим процессом на СП «Хмелинецкий сахарный завод» «АО «АПО «Аврора».

В частности:

1) теоретические, методологические и прикладные результаты исследования автора по иерахической идентификации квазилинейных окрестностных моделей и их применению к модели управления температурным параметрами стадии диффузии производства сахара анализировались инженерно-техническим персоналом СП «Хмелинецкий сахарный завод» при планировании мероприятий по совершенствованию управления и повышению технико-экономических показателей производства сахара заводом АО «АПО «Аврора»;

2) на основе выполненного анализа и опыта практического использования результатов диссертационной работы установлено, что предлагаемые методы работоспособны и позволяют обеспечивать повышение эффективности управлении технологическим процессом стадии диффузии производства сахара применительно к СП «Хмелинецкий сахарный завод».

Наиболее целесообразным является использование разработанных методов иерархической идентификации при уточнении и совершенствовании существующих математических моделей сложных производственных процессов.

Директор свеклосахарного производства
АО «АПО «Аврора» С.Н. Зобова

УТВЕРЖДАЮ
Проректор по учебной работе
ФГБОУ ВО «Липецкий государственный технический университет»
д.ф.н., доцент
_____Полякова И.П.
«29» 09 2023 г.

СПРАВКА
об использовании в учебном процессе материалов, содержащихся в кандидатской диссертации Щербакова Артема Петровича «Разработка методов и алгоритмов рекуррентной идентификации на основе иерархических окрестностных моделей».

Настоящей справкой удостоверяется, что результаты диссертационной работы на соискание ученой степени кандидата технических наук по специальности 1.2.2 «Математическое моделирование, численные методы и комплексы программ», а именно:

иерархические окрестностные структуры; иерархические окрестностные модели; алгоритмы рекуррентной иерархической идентификации регрессионных моделей на основе аппроксимации, кластеризации и трихотомии невязок промежуточных моделей, отличающиеся использованием иерархической кластеризации и иерархического разбиения обучающих входных данных и позволяющие улучшать точность аппроксимации; схема проверки адекватности иерархических моделей, отличающаяся анализом остаточных данных промежуточных моделей, упорядоченных в соответствии с возрастанием обучающих выходов; комплекс программ, реализующих алгоритмы иерархической идентификации, отличающийся наличием модуля вычисления ближайшего обучающего кортежа

используются в учебном процессе федерального государственного бюджетного учреждения высшего образования «Липецкий государственный технический университет» в рамках образовательной программы по направлению 01.03.03 «Механика и математическое моделирование» при выполнении индивидуальных заданий по дисциплине «Моделирование систем», а также подготовке выпускных квалификационных работ.

Использование результатов диссертационной работы обсуждено на заседании кафедры высшей математики от «19» сентября 2023 г., протокол № 2.

Начальник отдела по науке П.А. Кровопусков

Декан физико-технологического факультета
к.т.н., доцент И.А. Коваленко

Заведующий кафедрой высшей математики
д.т.н., профессор А.М. Шмырин

Приложение 2

Свидетельства о государственной регистрации программ для ЭВМ

РОССИЙСКАЯ ФЕДЕРАЦИЯ

СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2015614243

Расчет корреляционных размерностей и доли случайного хаоса по данным временного ряда

Правообладатель: ***Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Липецкий государственный технический университет» (ФГБОУ ВПО ЛГТУ) (RU)***

Авторы: ***Шмырин Анатолий Михайлович (RU), Щербаков Артем Петрович (RU), Седых Ирина Александровна (RU), Ярцев Алексей Геннадьевич (RU)***

Заявка № **2015610763**
Дата поступления **13 февраля 2015 г.**
Дата государственной регистрации
в Реестре программ для ЭВМ ***10 апреля 2015 г.***

Врио руководителя Федеральной службы по интеллектуальной собственности

Л.Л. Кирий

РОССИЙСКАЯ ФЕДЕРАЦИЯ

СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2015610766

Получение общего параметрического уравнения окрестностной модели

Правообладатель: ***Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Липецкий государственный технический университет» (ФГБОУ ВПО ЛГТУ) (RU)***

Авторы: ***Ярцев Алексей Геннадьевич (RU), Шмырин Анатолий Михайлович (RU), Щербаков Артем Петрович (RU), Седых Ирина Александровна (RU)***

Заявка № **2014661696**

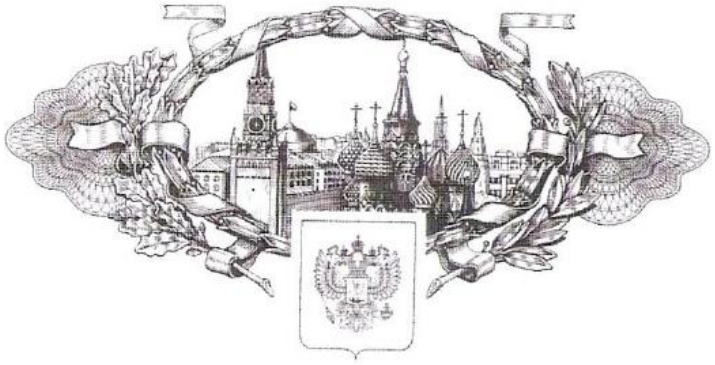
Дата поступления **19 ноября 2014 г.**

Дата государственной регистрации в Реестре программ для ЭВМ ***16 января 2015 г.***

Врио руководителя Федеральной службы по интеллектуальной собственности

Л.Л. Кирий

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2015617475

Исследование и решение систем методом ортонормализации Грама-Шмидта

Правообладатель: ***Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Липецкий государственный технический университет» (ФГБОУ ВПО ЛГТУ) (RU)***

Авторы: ***Семёнова Вера Васильевна (RU), Шмырин Анатолий Михайлович (RU), Седых Ирина Александровна (RU), Щербаков Артем Петрович (RU), Ярцев Алексей Геннадьевич (RU)***

Заявка № **2015612504**

Дата поступления **01 апреля 2015 г.**

Дата государственной регистрации в Реестре программ для ЭВМ ***10 июля 2015 г.***

Врио руководителя Федеральной службы по интеллектуальной собственности

Л.Л. Кирий