МИНОБРНАУКИ РОССИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«НОВОСИБИРСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ» (НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ, НГУ)

Факультет информационных технологий

Кафедра систем информатики

Направление подготовки: 230100 ИНФОРМАТИКА И ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА

Магистерская программа: Компьютерное моделирование

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

Интеллектуальная система анализа геолого-промысловых данных

Татарников Вадим Владимирович

Тема диссертации утверждена распоряжением по НГУ № 532 от «14» декабря 2012г.

Тема диссертации скорректирована распоряжением по НГУ № 205 от «8» мая 2014г.

«К защите допущена»

Заведующий кафедрой,

д.ф.-м.н., профессор

Лаврентьев М. М. /……..
(фамилия , И., О.) (подпись, МП)

«……»……20…г.

Научный руководитель

Главный научный сотрудник ИМ СО РАН,

д. т. н., профессор

Загоруйко Н. Г. /……...
(фамилия , И., О.) (подпись, МП)

«……»……20…г.

Дата защиты: «……»……20…г.

Автор: Татарников В. В. /……...
(фамилия , И., О.) (подпись)

Новосибирск, 2014г.

МИНОБРНАУКИ РОССИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «НОВОСИБИРСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ» (НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ, НГУ)

Факультет информационных технологий

Кафедра систем информатики

Направление подготовки: 230100 ИНФОРМАТИКА И ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА

Магистерская программа: Компьютерное моделирование

УТВЕРЖДАЮ

Зав. кафедрой……
(фамилия, И., О.)

…….
(подпись, МП)

«……»……20…г.

ЗАДАНИЕ

НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ

МАГИСТЕРСКУЮ ДИССЕРТАЦИЮ

Студенту: Татарникову Вадиму Владимировичу

Тема: Интеллектуальная система анализа геолого-промысловых данных

Исходные данные (или цель работы): Разработка и исследование методов заполнения пробелов в геолого-промысловых данных. Разработка системы анализа геолого-промысловых данных направленной на снижение степени их неопределённости путём заполнения пробелов, поиска и исправления ошибок.

Структурные части работы: Исследование методов заполнения пробелов в таблицах «объект-свойство». Разработка методов заполнения пробелов в кубах данных. Разработка интеллектуальной системы.

Научный руководитель

Главный научный
сотрудник ИМ СО РАН,
д. т. н., профессор

Загоруйко Н. Г. /……..
(фамилия , И., О.) / (подпись)

«…»……20…г.

Задание принял к исполнению

Татарников В. В. /……..
(ФИО студента) / (подпись)

«…»……20…г.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Мировая нефтегазовая индустрия имеет в настоящее время набор характерных проблем, а именно: падающая добыча, рост издержек, усложнение географических условий добычи, ухудшение качества запасов углеводородов, нехватка опытного персонала и высокая степень неопределённости данных, используемых для принятия решений. Одним из путей решения этих проблем является внедрение новой техники и технологий, в том числе автоматизация и информатизация производственных процессов. В частности, автоматизация обработки и анализа промысловых данных нефтегазовых месторождений, т.е. данных, получаемых в результате измерений, проводимых при эксплуатации нефтегазовых скважин. Таким образом, актуальным направлением становится разработка и внедрение интеллектуальных систем анализа геолого-промысловых данных и управления месторождением. Цель данной работы – создание программной системы, направленной снижение степени неопределённости геолого-промысловых данных.

Подобные системы предъявляют высокие требования к полноте и целостности промысловых данных по объёмному или массовому потоку каждого компонента трехфазной смеси (вода, нефть, газ). То есть, основой для любой системы интеллектуального управления месторождением является установка расходомеров. В настоящее время более 70% скважинного фонда РФ состоит из низкодебитных, сильнообводненных скважин, находящихся на поздней стадии эксплуатации. На такие скважины экономически нецелесообразно, а иногда и технически невозможно, устанавливать дорогостоящие расходометрические решения. Необходимость же установки расходомеров обусловлена как целями повышения рентабельности добычи и решения вышеописанных задач, так и законодательно (ГОСТ Р 8.615 - 2005, обязывающий проводить замеры дебита по нефти и другим параметрам по каждой скважине)

Сейчас решения расходометрии для низкопродуктивных скважин представляют собой автоматическую групповую замерную установку в случае кустовой организации скважин или передвижной расходомер в случае, если скважины удалены друг от друга. Подобные системы оснащаются входным многопортовым гидравлическим переключателем, что позволяет периодически замерять продуктивность каждой подключенной скважины (до 16 шт.). Однако, их недостатком является выпадение целых интервалов с данными по каждой скважине в то время, пока переключатель «перебирает» все остальные скважины куста, и погрешность из-за перетоков. В результате, примерно зная, какие дебиты должны быть на каждой скважине куста, специалист вынужден вручную расписывать групповой замер. Естественно, при таком подходе решение

вышеописанных задач интеллектуального управления разработкой месторождения невозможно.

В работах [1,2,3] предложена и обоснована целесообразность применения в данной ситуации замерной схемы переключения по событиям (рис.1). Данная схема требует установки на каждую скважину по одному устройству замера параметров потока и только одного высокоточного расходомера для групповых замеров дебита скважин. Тогда в любой момент времени точным расходомером измеряется только одна скважина, а дебит остальных определяется при помощи некоторого приближённого метода, точность которого существенно ниже, чем точность расходомера, однако всё ещё позволяет регистрировать гидродинамические события. По возникновению такого события на одной из скважин, управляющий узел переключает расходомер на эту скважину.

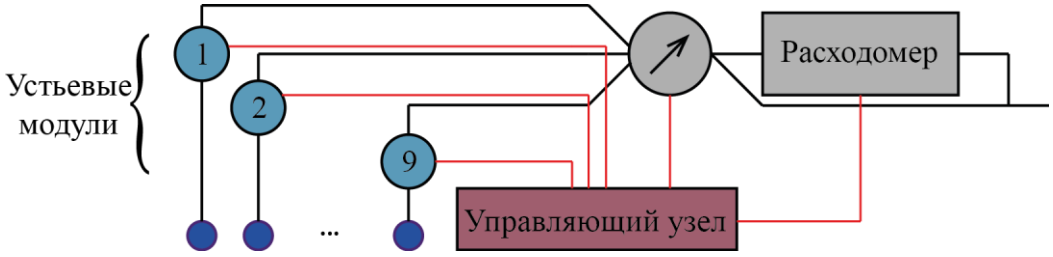


Рис. 1. Схема переключения по событиям.

В исследованных работах обоснована целесообразность применения такой схемы для мониторинга куста скважин, поскольку её применение снижает износ оборудования и даёт более точный учёт продуктивности скважин. На практике проблема неопределённости возникает и при сборе данных с устройств замера параметров потока: технические неполадки канала связи, низкие температуры и др. факторы вносят ошибки и пробелы в данные, использующиеся для дальнейших расчётов. Решить эту проблему или снизить её влияние позволит предобработка: поиск ошибок, и заполнение пробелов.

Для достижения поставленной цели необходимо разработать программную систему обеспечивающую сбор, хранение и обработку данных мониторинга скважин. Среди прочих особенностей, система должна иметь возможность обнаруживать ошибки и аномальное поведение, т.е. выделять «события», и заполнять пробелы, что потребует предварительного анализа существующих алгоритмов для решения обозначенных проблем. Кроме того необходимо обеспечить расширяемость системы путём реализации возможности добавления новых измеряемых параметров и алгоритмов расчёта и фильтрации.

В настоящей работе приведено описание реализации такой системы, а также, помимо стандартных решений, предложен собственный алгоритм заполнения пробелов в геолого-промысловых данных и критерии ошибки на его основе. Новизна разработанной

системы заключается в применении оригинальной схемы учёта продуктивности скважин, сочетание использования гидродинамических моделей и методов анализа данных, а также применение алгоритма собственной разработки, использующего функцию конкурентного сходства(FRiS-функцию)[4] и метод вычисления компактности на её основе[5].

1 РАЗРАБОТКА ИНТЕЛЛЕКТУАЛЬНОЙ СИСТЕМЫ

Для решения поставленной задачи, т.е. при разработке интеллектуальной системы, необходимо собрать и проанализировать требования к продуктам такого рода. Кроме того, помимо обнаружения ошибок, заполнения пробелов и выделения событий, необходимо реализовать широкий спектр других функций, например, визуализация и редактирование данных, генерация отчётов. Необходимо сделать систему расширяемой и учесть аспекты безопасности.

1.1 Анализ требований к системе

1.1.1 Хранение данных

К данным, которыми оперирует система, относятся: замеры произвольного числа параметров, произвольных скважин в некоторые моменты времени, некоторые расчётные данные на их основе, данные кустовых замеров, данные о событиях, системные настройки. Данные с измерительных систем, как правило, хранятся отдельно в БД либо файле. Поскольку система сама по себе является источником данных (расчёты и события) необходимо обеспечить их хранение. Кроме того, нужно учитывать ситуацию, при которой чтение из БД с замерами возможно, но не возможна их модификация, а также добавление своих данных. Приведённые соображения позволяют констатировать необходимость разворачивания собственной инфраструктуры БД. Это решение приводит к необходимости организовать автоматическую(по расписанию) синхронизацию баз данных, но предоставляет свободу в выборе схемы данных, максимально отвечающей требованиям задач по их обработке. Тем не менее, заранее необходимо предусмотреть возможность появления других скважин и измеряемых/расчётных параметров в системе.

1.1.2 Обработка данных

Функции обработки данных с помощью системы можно разделить на две группы: обработка без участия пользователя и обработка с участием пользователя. К обработке данных системой без участия пользователя можно отнести: поиск и исправление ошибок(пробелов) в измеряемых параметрах, расчёт параметров, выделение на их основе событий. Данный функционал должен быть расширяем и адаптируем как к изменениям списка скважин и списка параметров, так и к изменениям содержательной части алгоритмов обработки данных. Другими словами, каждый метод расчётов и метод поиска ошибок и выделения событий должен быть выполнен в виде подключаемого модуля, что позволит модифицировать существующие и разрабатывать новые методы отдельно от системы. К обработке данных с участием пользователя можно отнести: ручной импорт

данных, ручное редактирование, визуализацию данных, генерацию отчётов. Реализация перечисленных функций подразумевает создание графического интерфейса пользователя.

1.1.3 Безопасность

Поскольку система работает с данными, представляющими коммерческую тайну, при разработке и внедрении необходимо учесть необходимость ограничить доступ не авторизованных лиц к исходным данным и результатам их обработки. Данное требование может быть реализовано путём разработки системы пользователей с различными правами доступа: право на чтение исходных данных и событий и генерация отчётов, право на ручное редактирование и ручной импорт данных в систему, и так далее, вплоть до права на редактирование пользователей.

1.2 Реализация

После проведения анализа требований к разрабатываемой системе в качестве средства разработки был выбран язык программирования C# для платформы .NET 4.5. Концептуально систему можно разделить на несколько крупных, слабо зависящих друг от друга компонентов: модуль хранения данных, модуль синхронизации и расчётов, модуль редактирования и визуализации данных, и расширяемый набор методов расчёта и выявления событий(рис.2).

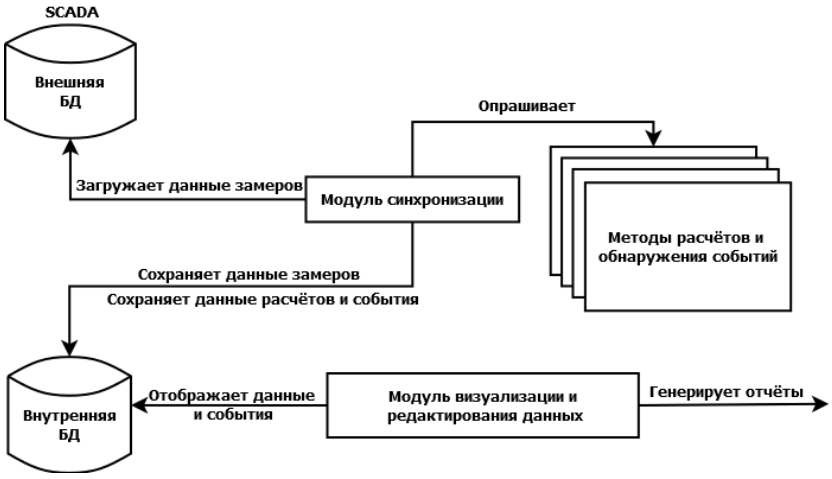


Рис. 2. Концептуальная схема взаимодействия модулей.

Графический интерфейс модуля редактирования и визуализации данных реализован при помощи библиотек WinForms и DevExpress(рис.3).

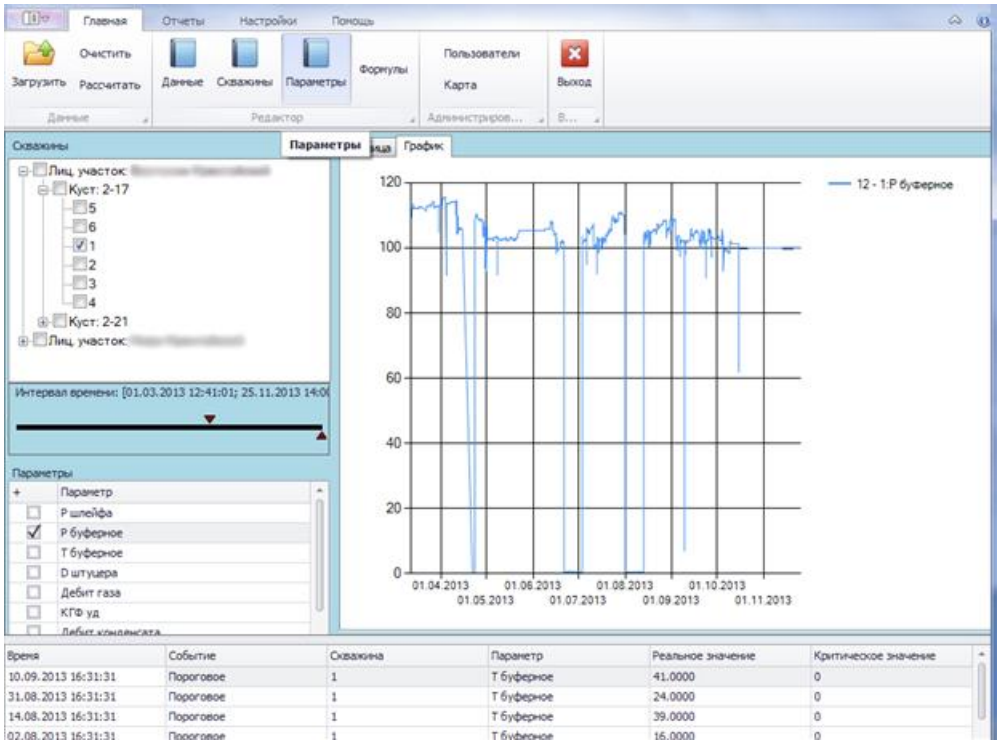


Рис. 3. Окно модуля редактирования и визуализации данных. Выбранные параметры, выбранных скважин, за указанный период времени отображаются на графике. На нижней панели обновляемый список событий.

1.2.1 Хранение данных

Для обеспечения требований к организации хранения данных в качестве СУБД выбрана MS SQL Server. Разработана схема данных охватывающая такие сущности как скважина, куст, лицензионный участок, измеряемый параметр, расчётный параметр, замер, событие, пользователь и другие(рис.4).

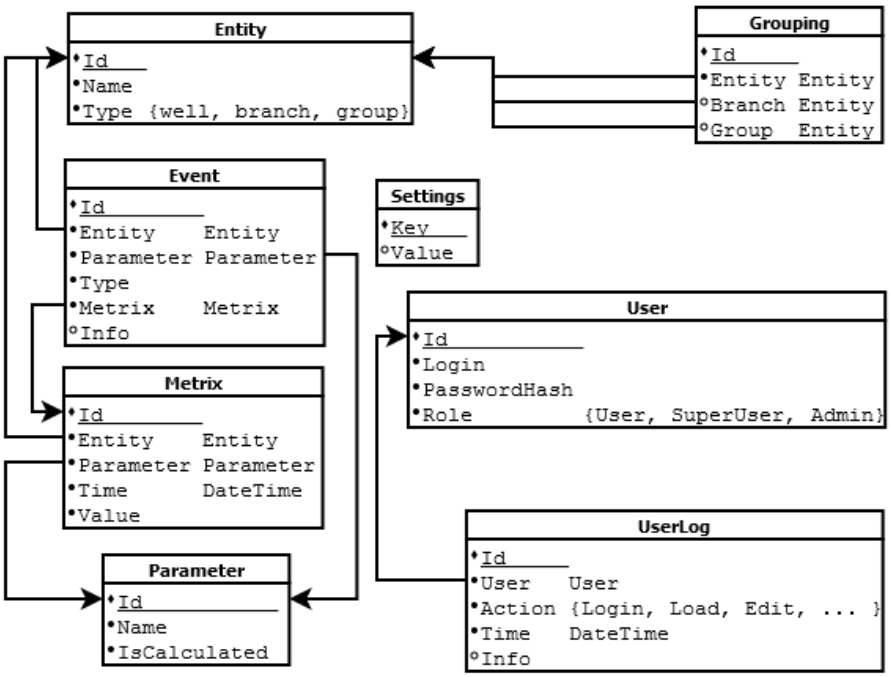


Рис. 4. Схема организации основных сущностей в БД.

Данные замеров и расчётные данные(Metrix) и представлены в виде записей Entity-Parameter-Time-Value, где Entity соответствует некоторой сущности, способной порождать данные, в данном случае речь идёт о скважинах, кустах скважин и лицензионных участках, иерархия которых задаётся посредством таблицы Grouping. Объекты Parameter описывают измеряемые и расчётные параметры. Выбранная схема данных позволяет хранить любое количество параметров для каждой скважины без избыточности, а так же позволяет группировать скважины по кустам и лицензионным участкам, которые в свою очередь тоже могут участвовать в расчётах и мониторинге как полноценные объекты. Например, для реализации, описанной во введении схемы переключения по событиям, скважины порождают записи Metrix с параметрами, измеряемыми устьевыми модулями, а объект соответствующий кусту порождает записи Metrix с дебитом, который измеряется расходомером. События хранятся в таблице Events и привязаны к объекту, параметру и времени, а также содержат дополнительное поле с информацией для интерпретации.

Каждому пользователю системы соответствует запись в таблице User, каждый пользователь состоит в одной из групп, наделённых соответствующими правами. Действия пользователя записываются в таблицу UserLog. Настройки системы, а также настройки методов хранятся в таблице Settings в виде пар ключ-значение.

Синхронизация БД с замерами и БД системы осуществляется посредством модуля синхронизации и расчётов, выполненного в виде службы(Windows Service), которая по расписанию выполняет импорт данных замеров из системы SCADA, применяет зарегистрированные в системе фильтры для поиска и исправления ошибок, выполняет

необходимые расчёты и применяет методы поиска событий. Отображение схемы данных внешней БД на БД конфигурируется с помощью файлов в формате XML. При необходимости добавить соединение с новой внешней системой или в случае изменения местоположения какой-либо из задействованных баз данных, файлы могут быть отредактированы администратором системы. Реализована возможность ручного импорта данных из файлов в формате XLS(рис.5).

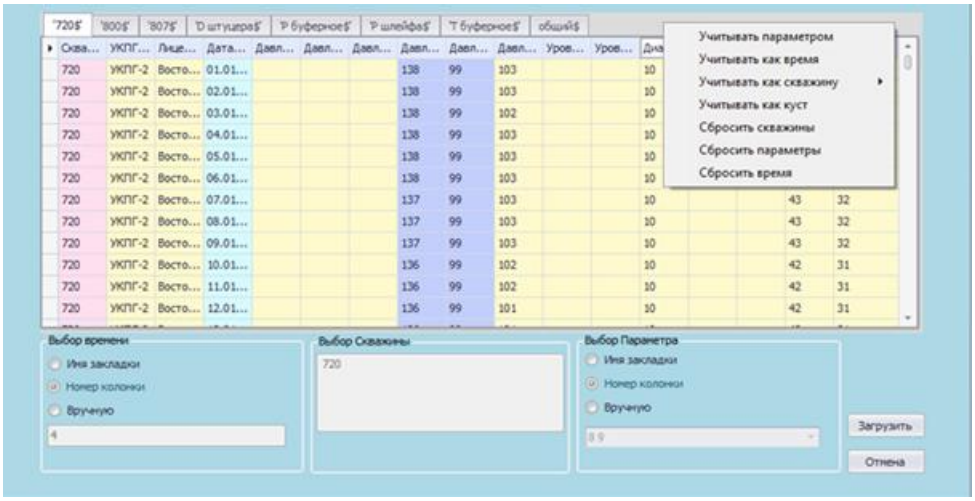


Рис. 5. Окно ручного импорта данных из XLS. Цветом выделяются столбцы с именем скважины, временем замера и параметрами, из которых будут сформированы объекты Metrix.

1.2.2 Обработка данных

Операции обработки данных не требующие участия пользователя вызываются по расписанию при помощи собственной службы, описанной в предыдущем пункте. Методы для фильтрации и выделения событий представляют собой библиотеки DLL с реализованными в них интерфейсами EventChecker:

```
public interface EventChecker
{
    string Name { get; }
    string Help { get; }
    string Version { get; }

    …

    Event[] findEvents(DateTime d, DataHelper helper);
}
```

Для регистрации методов определения событий служба опрашивает все файлы DLL в специально отведённой директории на соответствие интерфейсу EventChecker и извлекает информацию о фильтре: имя, которое используется для обозначения событий, краткое описание фильтра и его версию. Для выделения событий при получении новых данных служба взывает метод `findEvents` у всех зарегистрированных методов, передавая момент времени, который нужно проверить, и объект для доступа к данным. Результат

работы методов – выделенные события – служба сохраняет. Следует отметить, что методы поиска ошибок также представляют собой реализации EventChecker, поскольку ошибка или некомплектность в данных – тоже событие, которое требует принятия решения. Реализованы методы выделения событий при изменении параметра на величину большую заданного значения, описанный в [2,3] метод на основе линейной регрессии и доверительного конуса, а так же собственный метод обнаружения ошибок в кубах геолого-промысловых данных, подробно описанный в главе 2. Параметры методов выделения событий редактируются в настройках. Аналогично методам поиска ошибок и выделения событий устроены подключаемые модули с формулами для расчётов:

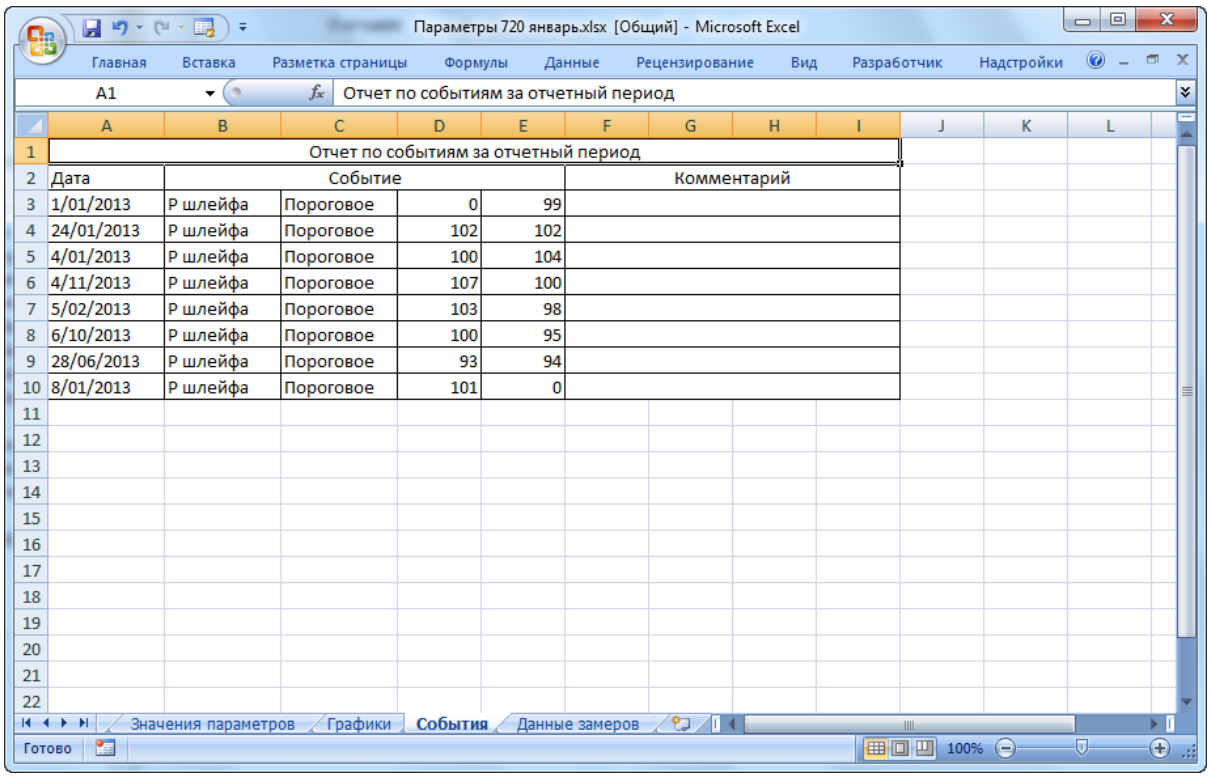
```
public interface Formula
{
    string Name { get; }
    string Help { get; }
    string Version { get; }

    …

    Metrix[] calculate(DateTime d, DataHelper helper);
}
```

Метод `calculate` отвечает за вычисление значения расчётного параметра в определённый момент времени. Интерфейс Formula, в частности, реализуется методом пересчёта данных устьевых замеров в поскважинные дебиты при помощи гидродинамической модели.

Операции обработки данных требующие непосредственного участия пользователя вызываются через графический интерфейс(рис.3,5). Разработанный модуль редактирования и визуализации данных позволяет просматривать данные в виде сводных таблиц и графиков, просматривать события и сохранять данные в виде отчётов в файлы XLS с несколькими листами данных(рис.6).



	A	B	C	D	E	F
1	Отчет по событиям за отчетный период					
2	Дата	Событие				Комментарий
3	1/01/2013	Р шлейфа	Пороговое	0	99	
4	24/01/2013	Р шлейфа	Пороговое	102	102	
5	4/01/2013	Р шлейфа	Пороговое	100	104	
6	4/11/2013	Р шлейфа	Пороговое	107	100	
7	5/02/2013	Р шлейфа	Пороговое	103	98	
8	6/10/2013	Р шлейфа	Пороговое	100	95	
9	28/06/2013	Р шлейфа	Пороговое	93	94	
10	8/01/2013	Р шлейфа	Пороговое	101	0	

Рис.6. Пример отчёта, созданного системой. На разных листах представлены события за отчётный период, данные замеров и графики.

При наличии соответствующих прав доступа, интерфейс позволяет вручную редактировать данные, параметры и объекты.

1.2.3 Безопасность

Для того чтобы ограничить доступ неавторизованных лиц к системе, разработана система пользователей. Для доступа к системе всем пользователям необходимо ввести логин и пароль, регистрацию новых пользователей осуществляет администратор системы. По полномочиям в системе пользователи разделены на три группы: «User», «SuperUser» и «Admin», полномочия групп приведены в таблице 1.

Таблица 1.

Разделение полномочий пользователей по группам

Действие в системе	**Группа пользователей**		
	User	SuperUser	Admin
Авторизация	+	+	+
Просмотр событий и данных, генерация отчётов	+	+	+
Ручная загрузка и редактирование данных	-	+	+
Редактирование настроек	-	+	+
Редактирование параметров и скважин	-	-	+
Редактирование пользователей и просмотр журнала	-	-	+

Кроме того, все действия пользователей в системе заносятся в журнал и доступны администратору системы для просмотра.

1.3 Результаты

В настоящей главе проведён анализ требований к интеллектуальной системе анализа геолого-промысловых данных, проведена декомпозиция на модули, учитывающая специфику задачи и описана реализация компонентов такой системы. Новизна разработанной системы заключается в практической реализации совместного применении гидродинамической модели и методов анализа данных для выделения событий, и практической реализации собственного метода обнаружения ошибок.

2 АЛГОРИТМ ЗАПОЛНЕНИЯ ПРОБЕЛОВ В ГЕОЛОГО-ПРОМЫСЛОВЫХ ДАННЫХ

2.1 Постановка задачи

Пусть имеется некоторое множество скважин $a \in A$, на каждой из которых измеряется некоторое множество параметров $x \in X$ в моменты времени $t \in T$. Таким образом, рассматриваемый набор данных образует «куб» $\langle A, X, T\rangle$, каждый элемент которого однозначно определяется тройкой $\langle a_i, x_j, t_k\rangle$ для некоторых $a_i \in A,\ x_j \in X,\ t_k \in T$. Сечением $\langle a_i, X, T\rangle$ куба назовём множество элементов $\{\langle a_i, x, t\rangle \mid x \in X, t \in T\}$, аналогичным образом определяются сечения $\langle A, x_j, T\rangle$ и $\langle A, X, t_k\rangle$. Иногда для краткости будем обозначать сечение символом соответствующего ему объекта a_i, признака x_j или момента времени t_k.

Пусть все элементы куба $\langle A, X, T\rangle$ определены, но необходимо проверить, не содержат ли они случайных или умышленных грубых ошибок. Для этого необходимо сделать ряд предположений о закономерностях в данных, т.е. ввести функции $Q(a, x, t)$, прогнозирующие значение элемента $\langle a, x, t\rangle_q = q(a, x, t), q \in Q$. Пусть элемент $\langle a, x, t\rangle'_i$ получен при помощи метода $q_i \in Q$, а $\langle a, x, t\rangle$ его истинное значение. Тогда ошибку $D(q_i)$ метода прогнозирования q_i определим так:

$$D(q_i) = \frac{1}{|A||X||T|}\sum_{a\in A}\sum_{x\in X}\sum_{t\in T} d_{q_i}(a, x, t),$$

$$d_{q_i}(a, x, t) = \left|\frac{\langle a, x, t\rangle - \langle a, x, t\rangle'_{q_i}}{\langle a, x, t\rangle}\right| \times 100\%.$$

Задача выбора оптимальной прогнозирующей функции заключается в выборе такого метода $q \in Q$, обеспечивающего минимум величины суммарной ошибки:

$$q = \arg\min_{\mathrm{q}\in\mathrm{Q}} D(q)$$

Будем говорить, что куб $\langle A, X, T\rangle$ содержит пробел в $\langle a_0, x_0, t_0\rangle$, если его значение отсутствует в рассматриваемом наборе данных. Задача заполнения пробела $\langle a_0, x_0, t_0\rangle$ в кубе $\langle A, X, T\rangle$ заключается в отыскании его значения. Поскольку истинное значение элемента не известно и получить ошибку в явном виде невозможно, в качестве решения целесообразно взять $\langle a_0, x_0, t_0\rangle' = q(a_0, x_0, t_0)$, где q – оптимальная прогнозирующая функция.

2.2 Анализ методов решения

Существующие методы анализа таблиц пробелами [6,7,8] можно условно разделить на простые и сложные. К простым методам относятся: интерполяция и заполнение скользящим средним. Среди особенностей данных методов можно выделить простоту

применения и локальность, т.е. пробел заполняется значением, вычисленным на основе элементов из некоторой окрестности (по времени или среди объектов) рассматриваемого элемента. Явный недостаток этих методов заключается в том, что они не учитывают возможные зависимости между объектами и их свойствами. Кроме того, вследствие усреднения целевой характеристики возможны сглаживания особенностей кривых. Лишены этих недостатков сложные методы решения обозначенной задачи, такие как: регрессия и EM-алгоритм[9]. Задачей регрессии является аппроксимация с минимальной ошибкой набора данных кривой (или поверхностью) из заданного класса (линейные, квадратичные и другие). EM-алгоритм подбирает модель распределения данных на основе оценок максимального правдоподобия. Получив модель, известные значения в неё можно подставить, чтобы восстановить пробел. Однако данные методы используются в предположении о фиксированном виде зависимости, которая реализована на всех объектах в каждый момент времени. Использование этого предположения сужает класс задач, в которых применимы вышеперечисленные алгоритмы, либо требует дополнительной предобработки данных: кластеризации, уменьшения размерности пространства признаков, выделения трендов.

В работе [10] впервые был предложен локальный алгоритм, оценивающий зависимости в данных в некоторой окрестности предсказываемого объекта. Алгоритм разработан для двумерных таблиц «объект-свойство» $\langle A, X\rangle$. Для заполнения пробела $\langle a_0, x_0\rangle$ алгоритм строит подтаблицу $\langle A', X'\rangle$, названную компетентной, из строк и столбцов, наиболее близких с точки зрения Евклидового расстояния к $\langle a_0, X\rangle$ и $\langle A, x_0\rangle$. Значение $\langle a_0, x_0\rangle'$ оценивается при помощи линейной регрессии между столбцами x_0 и x_i:

$$\langle A', x_0\rangle = f\big(\langle A', x_j\rangle\big),$$

$$\langle a_0, x_0\rangle'_j = f\big(\langle a_0, x_j\rangle\big),$$

Далее полученные оценки усредняются:

$$\langle a_0, x_0\rangle' = \frac{1}{|X'|}\sum_{j=1}^{|X'|}\langle a_0, x_0\rangle'_j.$$

Аналогичным образом, значение $\langle a_0, x_0\rangle'$ можно оценить регрессией по строкам. Разработаны модификации алгоритма, использующие коэффициент линейной корреляции в качестве меры сходства столбцов, взвешенную сумму оценок при прогнозировании, а так же способы вычисления расстояния для разнотипных данных, например, измеренных в шкале наименований.

При формировании подматрицы алгоритм следует предположению об избыточности и локальной компактности в данных, которые означают, что для

восстановления пробела достаточно использовать подмножество строк, наиболее похожих на строку с пробелом, и подмножество столбцов, наиболее похожих на столбец с пробелом. Для применения описанного подхода в анализе кубов данных алгоритм необходимо модифицировать таким образом, чтобы формировать не подматрицу $\langle A', X' \rangle$, а подкуб $\langle A', X', T' \rangle$, оперируя при этом не отдельными столбцами и строками, а сечениями. Необходимо предложить меру сходства сечений, а также способ оценки компактности подмножеств сечений A', X', T' и подкуба $\langle A', X', T' \rangle$ в целом.

2.3 Компактность

Определим расстояние между двумя произвольными сечениями $\langle a_i, X, T \rangle$ и $\langle a_j, X, T \rangle$ через Евклидово расстояние:

$$r^2(a_i, a_j) = \sum_{x \in X} \sum_{t \in T} \left(\langle a_i, X, T \rangle - \langle a_j, X, T \rangle \right)^2 .$$

Вычислив таким образом расстояния между целевым сечением a_0 и произвольными a_i, a_j $r(a_0, a_i)$ и $r(a_0, a_j)$, можно ответить на вопрос: какое из сечений больше похоже на целевое. Для того, чтобы получить количественную оценку сходства сечений, предположим, что $r(a_0, a_i) < r(a_0, a_j)$, и применим функцию конкурентного сходства(FRiS-функцию)[4]:

$$F(a_0, a_i | a_j) = \frac{r(a_i, a_j) - r(a_0, a_i)}{r(a_i, a_j) + r(a_0, a_i)}.$$

При совпадении сечений a_0 и a_i данная величина равна единице, по мере продвижения a_i к a_j она равномерно убывает, достигает нуля при $r(a_i, a_j) = r(a_0, a_i)$ и минус единицы при совпадении a_i с a_j. Таким образом, функция $F(a_0, a_i | a_j)$ является более информативным средством определения сходства сечений, чем $r(a_i, a_j)$.

Конкурентное сходство сечений позволяет также определить количественную меру компактности[5] множеств сечений A', X', T' подкуба $\langle A', X', T' \rangle$. Пусть a_0 – целевое сечение, $a^* = \arg\max_{a \in A} r(a_0, a)$ – наиболее удалённое от целевого. Медианой множества A' назовём виртуальный элемент $\bar{a}_0$: $r(a_0, \bar{a}_0) = \frac{1}{|A'|} \sum_{a \in A'} r(a_0, a)$, а медианой множества A/A' виртуальный элемент $\bar{a}^*$: $r(a_0, \bar{a}^*) = \frac{1}{|A \setminus A'|} \sum_{a \in A \setminus A'} r(a_0, a)$. Определим компактность A' как среднее значение конкурентного сходства сечений A' и медианы $\bar{a}_0$ в конкуренции с $\bar{a}^*$ при фиксированных X' и T':

$$\mathrm{C}_{X'T'}(A') = \frac{1}{|A'|} \sum_{a \in A'} F(\bar{a}_0, a | \bar{a}^*).$$

Чем выше плотность объектов внутри A' и чем дальше A' и A/A' отстоят друг от друга, тем больше величина компактности. Таким образом, отыскав максимум $\mathrm{C}(A')$, получим наиболее компактное множество сечений, похожих на a_0. Для вычисления максимума $\mathrm{C}(A')$ необходимо выбирать подмножества $A' \subset A$. Упорядочим сечения $a \in A$ по их расстоянию до a_0. Тогда A' и A/A' однозначно определяются номером граничного элемента a_i: $A'_i = \{a_1, \dots, a_i\}$, $A/A'_i = \{a_{i+1}, \dots, a_{|A|}\}$, что позволяет построить график $\mathrm{C}(A'_i)$ (рис.7) и найти $A'_{comp} = \arg\max_{A'_i} \mathrm{C}_{X'T'}(A'_i)$.

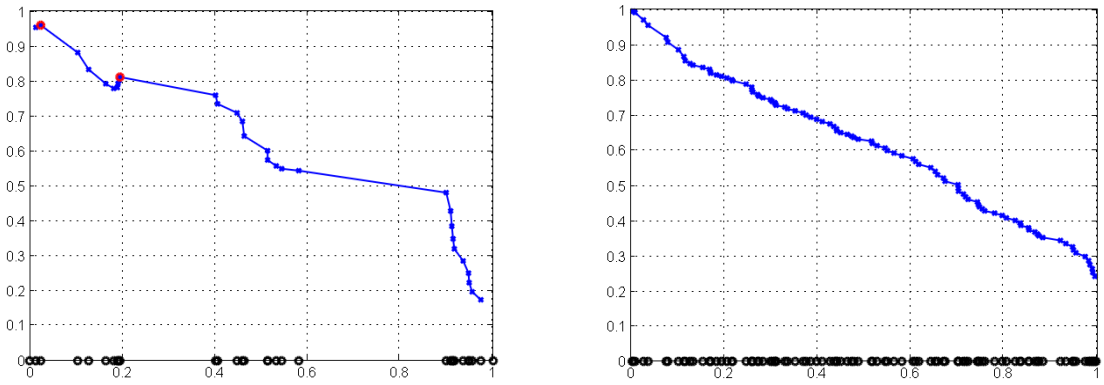


Рис. 7. График $\mathrm{C}(A'_i)$ *при компактном и равномерном распределении объектов на прямой* $r(a_0, a)$. *Красным выделены локальные максимумы.*

Оперируя сечениями $x \in X$ и $t \in T$, аналогичным образом определим компактности X' и T', а также компактность подкуба $\langle A', X', T' \rangle$ как среднюю компактность по трём направлениям:

$$\mathrm{C}_{A'T'}(X') = \frac{1}{|X'|} \sum_{x \in X'} F(\bar{x}_0, x | \bar{x}^*),$$

$$\mathrm{C}_{A'X'}(T') = \frac{1}{|T'|} \sum_{t \in T'} F(\bar{t}_0, t | \bar{t}^*),$$

$$\mathrm{C}\langle A', X', T' \rangle = \frac{\mathrm{C}(A') + \mathrm{C}(X') + \mathrm{C}(T')}{3}.$$

Таким образом, поиск компактного подкуба(КПК) заключается в решении следующей задачи:

$$C\langle A', X', T' \rangle \underset{\substack{A' \subset A \\ X' \subset X \\ T' \subset T}}{\longrightarrow} max.$$

2.4 Построение КПК и восстановление пробела

Решение задачи поиска компактного подкуба путём перебора всех возможных конфигураций сечений потребует порядка $|A'|! \times |X'|! \times |T'|!$ вычислений компактности,

поэтому целесообразно применять приближённый алгоритм. Рассмотрим следующий жадный алгоритм построения $\langle A', X', T' \rangle$:

1 $A' = \{a_0, a_1, \dots, a_{N^A_{min}}\}$, $X' = \{x_0, x_1, \dots, x_{N^X_{min}}\}$, $T' = \{t_0, t_1, \dots, t_{N^T_{min}}\}$.
2 **while** (**not** $\max C\langle A', X', T' \rangle$) **and** ($|A'| < N^A_{max}$ **or** $|X'| < N^X_{max}$ **or** $|T'| < N^T_{max}$) **do**
3 $\langle A', X', T' \rangle = Addition(\langle A', X', T' \rangle,\ \langle A, X, T \rangle)$;
4 **end**

Где процедура Addition определена так:

1 **def** Addition**:**
2 **for** $i = 1 \dots k_{add}$ **do**
3 $a_{add} = \arg\max_{a \in A/A'} F(\bar{a}_0, a|\bar{a}^*)$;
4 $x_{add} = \arg\max_{x \in X/X} F(\bar{x}_0, x|\bar{x}^*)$;
5 $t_{add} = \arg\max_{t \in T/T'} F(\bar{t}_0, t|\bar{t}^*)$;
6 $A' = A' \cup \{a_{add}\}$;
7 $X' = X' \cup \{x_{add}\}$;
8 $T' = T' \cup \{t_{add}\}$;
9 **end**
10 **return** $\langle A', X', T' \rangle$;

Данный алгоритм начинает формировать подкуб из некоторой начальной конфигурации сечений $\langle A', X', T' \rangle = \langle \{a_0, a_1, \dots, a_{N^A_{min}}\}, \{x_0, x_1, \dots, x_{N^X_{min}}\}, \{t_0, t_1, \dots, t_{N^T_{min}}\} \rangle$ и на каждом шаге наращивает по k_{add} сечений в каждом из направлений. Критериев остановки два: достижение локального максимума $C\langle A', X', T' \rangle$, и достижение предварительно заданных максимальных значений размеров $N^{A,X,T}_{max}$ подкуба, которые выбираются исходя из степени избыточности данных, чем она выше, тем большие ожидаются размеры у компактных подмножеств A', X', T'. Минимальный размер $N^{A,X,T}_{min}$ выбирается согласно требованиям выбранной прогнозирующей функции. Начальная конфигурация сечений может быть сформирована, например, путём выбора N^A_{min} ближайших сечений $\langle a, X, T \rangle$, N^X_{min} ближайших сечений $\langle A, x, T \rangle$ и N^T_{min} ближайших сечений $\langle A, X, t \rangle$, т.е. в оценке их близости к целевым сечениям участвуют все объекты A, свойства X и моменты времени T, тогда как дальше используются уже вошедшие в подкуб. Следует также отметить, что в описанном выше алгоритме и далее в этой работе при вычислении значений функций r, F и C члены с $\langle a_0, x_0, t_0 \rangle$ не учитываются, поскольку данное значение не определено.

Главный недостаток рассмотренного «жадного» алгоритма Addition состоит в том, что он локально оптимален и легко уводит решение от глобального результата. Кроме того, при предложенном способе выбора начальной конфигурации используются все сечения, в том числе и те, которые в конечном итоге не войдут в $\langle A', X', T' \rangle$. Чтобы избежать или смягчить влияние обозначенных проблем, дополним алгоритм процедурой исключения сечений Deletion:

1 **def** Deletion**:**
2 **for** $i = 1 \dots k_{del}$ **do**

3 $a_{del} = \arg\min_{a \in A'} F(\bar{a}_0, a|\bar{a}^*)$;
4 $x_{del} = \arg\min_{x \in X'} F(\bar{x}_0, x|\bar{x}^*)$;
5 $t_{del} = \arg\min_{t \in T'} F(\bar{t}_0, t|\bar{t}^*)$;
6 $A' = A'/\{a_{del}\}$;
7 $X' = X'/\{x_{del}\}$;
8 $T' = T'/\{t_{del}\}$;
9 **end**
10 **return** $\langle A', X', T' \rangle$;

Данная процедура исключает из $\langle A', X', T' \rangle$ k_{del} сечений с наименьшим значением F. Чередование шагов добавления k_{add} сечений и удаления k_{del} в каждой итерации позволяет составить аналог алгоритма AdDel для кубов данных:

1 $A' = \{a_0, a_1, \dots, a_{N^A_{min}}\}, X' = \{x_0, x_1, \dots, x_{N^X_{min}}\}, T' = \{t_0, t_1, \dots, t_{N^T_{min}}\}$.
2 **while** (**not** $\max C\langle A', X', T' \rangle$) **and** ($|A'| < N^A_{max}$ **or** $|X'| < N^X_{max}$ **or** $|T'| < N^T_{max}$) **do**
3 $\langle A', X', T' \rangle = Addition(\langle A', X', T' \rangle, \langle A, X, T \rangle)$;
4 $\langle A', X', T' \rangle = Deletion(\langle A', X', T' \rangle, \langle A, X, T \rangle)$;
5 **end**

При выборе $k_{del} < k_{add}$ на каждой итерации алгоритма размер подкуба $\langle A', X', T' \rangle$ увеличивается на $k_{add} - k_{del}$ по каждому направлению, т. е. выбор k_{del} и k_{add} позволяет управлять скоростью (числом итераций) формирования $\langle A', X', T' \rangle$. При выборе $k_{del} > k_{add}$ и инициализации $A' = A,\ X' = X,\ T' = T$ получим алгоритм последовательного исключения сечений.

При вычислении $C\langle A', X', T' \rangle$ на практике, на каждой итерации алгоритма в процедурах Addition и Deletion требуется вычисление медиан $\bar{a}_0,\ \bar{x}_0,\ \bar{t}_0,\ \bar{a}^*,\ \bar{x}^*,\ \bar{t}^*$ подмножеств сечений $A',\ X',\ T',\ A/A',\ X/X',\ T/T'$ соответственно, что сказывается на производительности при больших размерах исходных данных. Если предположить, что внутри компактного подмножества объекты распределены равномерно, то можно сосредоточиться не на поиске глобального максимума компактности, а на отыскании подмножества с равномерным распределением расстояний до целевых $a_0,\ x_0,\ t_0$. Это соображение позволяет модифицировать стратегию построения $\langle A', X', T' \rangle$:

1 $A' = A, X' = X, T' = T$.
2 **while** ($|A'| \geq N^A_{min}$ **or** $|X'| \geq N^X_{min}$ **or** $|T'| \geq N^T_{min}$) **do**
3 $a^* = \arg\max_{a \in A'} r(a_0, a)$;
4 $A'_> = \{a : F(a_0, a|a^*) > 0\}$;
5 $x^* = \arg\min_{x \in X'} r(x_0, x)$;
6 $X'_> = \{x : F(x_0, x|x^*) > 0\}$;
7 $t^* = \arg\max_{t \in T'} r(t_0, t)$;
8 $T'_> = \{t : F(t_0, t|t^*) > 0\}$;
9 **if** ($\frac{|A'_>|}{|A'|} > \alpha$) $A' = A'_>$; **end**
10 **if** ($\frac{|X'_>|}{|X'|} > \alpha$) $X' = X'_>$; **end**
11 **if** ($\frac{|T'_>|}{|T'|} > \alpha$) $T' = T'_>$; **end**

12 **if** $(\frac{|A'_>|}{|A'|} \leq \alpha$ **and** $\frac{|X'_>|}{|X'|} \leq \alpha$ **and** $\frac{|T'_>|}{|T'|} \leq \alpha)$ **break**; **end**

13 **end**

Приведённый алгоритм начинает с $\langle A', X', T' \rangle = \langle A, X, T \rangle$, на каждой итерации находит a^*, x^*, t^* и исключает сечения с отрицательным сходством. Для того чтобы контролировать скорость сокращения размера подкуба вводится коэффициент α. Если по одному из направлений количество сечений с отрицательным сходством слишком велико, т.е. $|A'_>| > \alpha|A'|$ и их исключение приведёт к значительному уменьшению размеров подкуба целесообразно не исключать сечения на этом шаге, а подождать следующего. Таким образом, выполнение этого условия по каждому из направлений останавливает работу алгоритма (рис.8). Равномерному распределению объектов соответствует $\alpha = 0.5$.

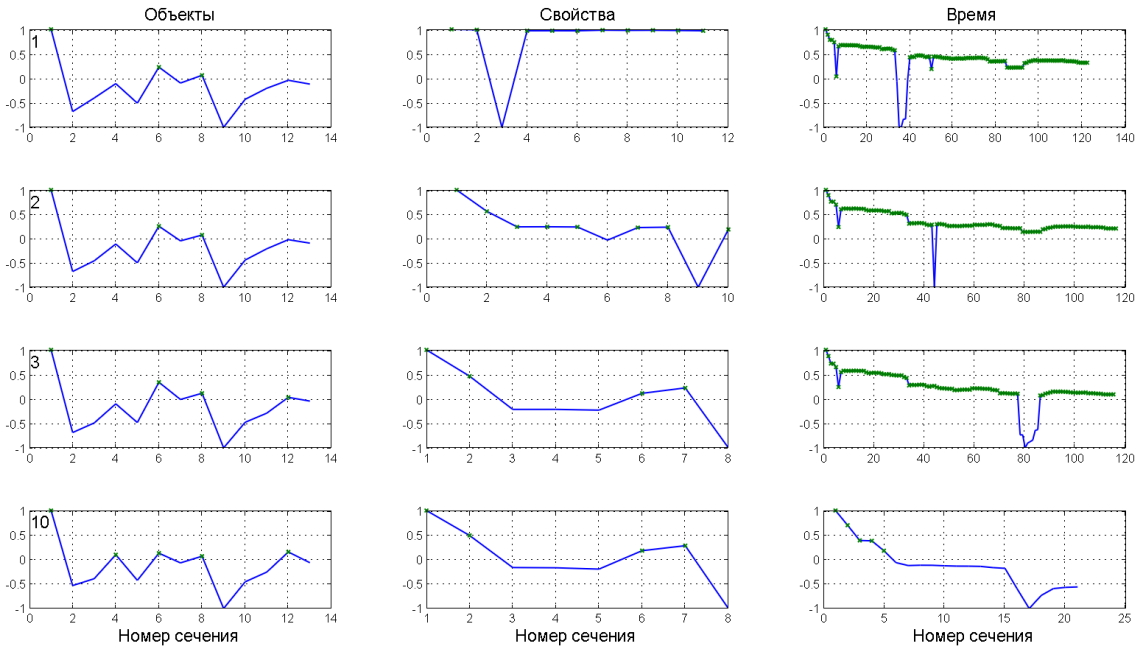


Рис. 8. Графики $F(a_0, a_i|a^*)$, $F(x_0, x_j|x^*)$, $F(t_0, t_k|t)$ *на разных итерациях алгоритма (номер итерации в углу). Выделены точки* $F_{i,j,k} > 0$.

Данный метод применим при высокой избыточности данных, а так же при больших размерах $\langle A, X, T \rangle$, в случаях, когда необходимо быстро сократить размеры $\langle A', X', T' \rangle$.

Получив компактный подкуб $\langle A', X', T' \rangle$ и опираясь на гипотезу локальной компактности, можно перейти к восстановлению значения элемента $\langle a_0, x_0, t_0 \rangle$ при помощи прогнозирующей функции q. Предложим несколько простых вариантов q. Усреднение целевой характеристики x_0 по времени:

$$q_{time}(a_0, x_0, t_0) = \frac{1}{|T'|} \sum_{t \in T'} \langle a_0, x_0, t \rangle.$$

В случае если сечения $t \in T'$ представляют собой некоторую окрестность t_0, результат будет эквивалентен применению алгоритма скользящего среднего. Усреднение целевой

характеристики x_0 по объектам в момент времени t_0 позволит найти среднее значение этой характеристики у «похожих» объектов:

$$q_{mean}(a_0, x_0, t_0) = \frac{1}{|A'|} \sum_{a \in A'} \langle a, x_0, t_0 \rangle.$$

Более полно раскрыть информацию о сходстве объектов и признаков позволит многомерная регрессия в сечении целевого объекта a_0:

$$q_{regr}(a_0, x_0, t_0) = \beta_{X'}(a_0, t_0),$$

$$MSE(\beta_{X'}) = \frac{1}{|T'|} \sum_{t \in T'} (\beta_{X'}(a_0, t) - \langle a_0, x_0, t \rangle)^2 \underset{\beta}{\rightarrow} min.$$

При этом вид её членов (линейные, полиномиальные и др.) выбирается на основании предположений о возможном характере зависимостей в данных. Дополнительным преимуществом использования регрессии является встроенная возможность оценить её ошибку, чем выше среднеквадратичное отклонение $MSE(\beta_{X'})$ модели $\beta_{X'}$, тем вероятней ожидать большую ошибку при прогнозировании $\langle a_0, x_0, t_0 \rangle$ с её помощью.

2.5 Применение на реальных данных

Разработанный алгоритм построения компактного подкуба и восстановления пропущенного значения был применён для оценки согласованности данных дебита с историей изменений параметров скважин. Данные представляют собой таблицы, где с фиксированным интервалом представлены показатели давления в системе, время работы скважины, процентное содержание воды, а также показания расходомеров (дебиты) каждой скважины. Данные не содержат пробелов и проверены экспертами на содержание ошибок. Для оценки точности работы алгоритма использовался метод перекрёстной проверки. Обозначим параметр «дебит» x_0. Далее каждый элемент $\langle a, x_0, t \rangle, a \in A, t \in T$ исходного куба положим неизвестным при известных остальных, построим компактный подкуб $\langle A', X', T' \rangle$, восстановим значение с помощью линейной регрессии в сечении целевой скважины $\langle a_0, x_0, t_0 \rangle' = q_{regr}(a_0, x_0, t_0)$ и сравним с известным значением $\langle a_0, x_0, t_0 \rangle$. Среднее значение относительной ошибки $D(q_{regr})$ в данном эксперименте составило 2%, гистограмма относительных ошибок $d_{q_i}(a, x_0, t)$ приведена на рисунке 9 слева. Отметим также, что велична $d_{regr}^{[0;10]} = \frac{|d_{regr}(a_0, x_0, t_0) > 10\%|}{|A||T|} \times 100\%$, т.е. соотношение элементов, относительная ошибка которых лежит в интервале $[0; 10]$, ко всем проверяемым элементам составило 96%. Результаты аналогичного эксперимента без построения подкубов $\langle A', X', T' \rangle$, т.е. при $\langle a_0, x_0, t_0 \rangle' = R(a_0, X, T)$ на рисунке 9 справа.

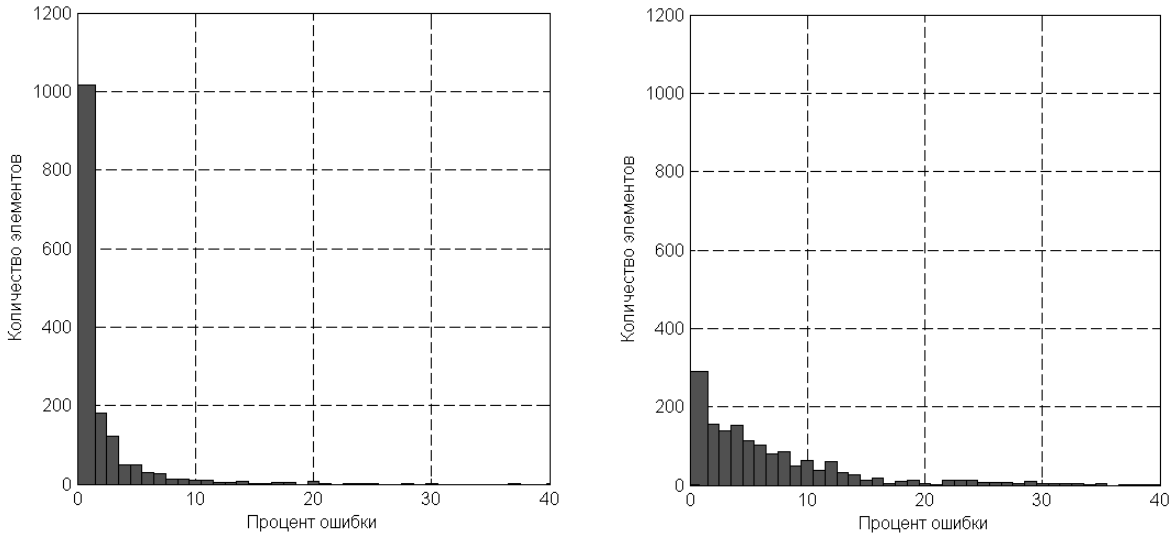


Рис. 9. Гистограммы относительной ошибки при перекрёстной проверке.

В эксперименте без построения подкубов, $D(q_{regr})$ составила 7%, а в интервале $[0; 10]$ находится уже 76% ячеек. Проверка дебита путём построения линейной регрессии кривой дебита от времени по каждой скважине в отдельности: $\langle a, x_0, T\rangle' = q_{trend}(a, x_0, T) = f_{a,x_0}(T)$ показала среднее значение относительной ошибки 5%, а в интервал $[0; 10]$ попало 85% исследуемых ячеек. Результаты экспериментов вынесены в таблицу 2.

Таблица 2.

Сравнение методов заполнения пробелов с помощью ошибки $D(q)$

Метод q	$\boldsymbol{D(q)}$	$\boldsymbol{d_q^{[0;10]}}$
q_{regr}	2%	96%
q_{regr} без построения КПК	7%	76%
q_{trend}	5%	85%

Таким образом, в данном эксперименте было установлено, что построение компактного подкуба улучшает точность перекрёстной проверки дебита при помощи регрессионной модели q_{regr}, что в контексте эксперимента делает её оптимальной прогнозирующей функцией. Кроме того, полученная в ходе проверки всего куба $\langle A, X, T\rangle$ статистическая информация может использоваться в качестве критерия оценки ошибок поступающих данных в будущем.

2.6 Обнаружение ошибок

Пусть куб данных $\langle A, X, T\rangle$ определён и не содержит ошибок, а для каждой характеристики $x \in X$ известна оптимальная прогнозирующая функция q, для которой

вычислено $d_q^{[0;\gamma_x]}$, для некоторых γ_x – пороговых значений. Тогда при поступлении новых данных $\langle A, X, t_{|T|+1}\rangle$, методом, описанным выше, необходимо вычислить $q\left(a, x, t_{|T|+1}\right), a \in A, x \in X$ и сравнить $d_q\left(a, x, t_{|T|+1}\right)$ с γ_x: если ошибка превысила пороговое значение, то на основании статистических данных можно говорить об ошибке в $\langle a, x, t_{|T|+1}\rangle$ с вероятностью $d_q^{[0;\gamma_x]}$. Периодический пересчёт $d_q^{[0;\gamma_x]}$ позволяет сделать этот критерий актуализируемым с течением времени и пополнением исходных данных.

2.7 Результаты

В данной главе разработан новый алгоритм заполнения пробелов в кубах геолого-промысловых данных, на его основе предложен адаптивный во времени критерий обнаружения ошибок. Новизна разработанного метода заполнения пробелов заключается в применении функции конкурентного сходства и критерия компактности на её основе. Практическая значимость метода заключается в возможности его применения для кубов данных любой природы, введение других видов расстояний r, использующих, например коэффициенты линейной или кросс-корелляции[11,12], или расстояний r для данных измеренных в разнотипных шкалах, представляет интерес для дальнейшего исследования. Алгоритм и критерий обнаружения ошибок реализован в разработанной системе в форме метода выделения событий, с настраиваемыми размерами куба данных.

ЗАКЛЮЧЕНИЕ

В рамках выполнения настоящей работы получены следующие результаты: разработана интеллектуальная система анализа и обработки геолого-промысловых данных. Отличительными особенностями системы являются слабая зависимость её компонентов друг от друга, адаптируемость к различным измерительным системам и схеме организации месторождения, а также возможность расширять список выделяемых событий собственными методами. Новизна разработанной системы заключается в практической реализации возможности совместного применения гидродинамической модели и методов анализа данных для выделения событий. Исследованы существующие методы заполнения пробелов в данных и разработан собственный метод, использующий новые результаты в области анализа данных. Разработанный метод может быть применён не только для анализа кубов геолого-промысловых данных, но кубов данных любой другой природы, кроме того, предложены возможности модификации алгоритма для анализа разнотипных кубов данных.

Разработанная система и алгоритмы выделения событий направлены на снижение степени неопределённости геолого-промысловых данных, используемых для принятия решений во время эксплуатации нефтегазовых месторождений.

ЛИТЕРАТУРА

1. Черемисин А. Н., Костюченко С. В., Торопецкий К. В., Рязанцев А. Э., Лукьянов Э. Е., Загоруйко Н. Г. Алгоритмы обработки результатов многофазной расходометрии в информационном обеспечении интеллектуального месторождения. // Нефтяное хозяйство. – 2013. – №6 – С. 98-10.
2. Ломухин А.Ю., Черемисин А.Н., Торопецкий К.В., Рязанцев А.Э. Интеллектуальная система распределённого мониторинга продуктивных параметров добывающих скважин. // Вестник ЦКР Роснедра. – 2013. – №4 – С .30-37.
3. Рязанцев А.Э., Бучинский С.В., Черемисин А.Н., Торопецкий К.В., Ломухин А.Ю. Количественная оценка погрешности различных методов замеров дебитов газоконденсатных скважин при инструментальном контроле технологических режимов. // Инженерная практика – 2013. – №6-7.
4. Zagoruiko, N. G., Borisova, I. A., Dyubanov, V. V., Kutnenko, O. A.: Methods of Recognition Based on the Function of Rival Similarity. // Pattern Recognition and Image Analysis, Vol. 18, No.1, 1–6 (2008).
5. Загоруйко Н. Г., Борисова И. А., Кутненко О. А., Дюбанов В. В. Построение сжатого описания данных с использованием функции конкурентного сходства. // Сибирский журнал индустриальной математики. Январь-март, 2013. Том XVI, № 1(53).
6. Buck S. F. A method of estimation of missing values in multivariate data // J. Roy. Statist. Soc. Ser. B. 1960. V. 22. P. 202-206.
7. Gleason T. C., Staelin R. A proposal for handling missing // Psychometrika. 1975. V.40. P. 229-252.
8. Frane G . M. Some simple procedure for handling missing values in multivariate analysis // Psychometrika. 1976. V. 41. P. 409-415.
9. Dempster A. P., Laird N. M., Rubin D. B. Maximum likelihood from incomplete data via the EM-algorithm // J. Roy. Statist. Soc. Ser. B. 1977. V. 39. P. 1-38.
10. Загоруйко Н.Г., Елкина В. Н., Темиркаев В. С. Алгоритм заполнения пропусков в эмпирических таблицах (алгоритм ZET) // Эмпирическое предсказание и распознавание образов. Вычислительные системы. 1975. Вып. 61. С. 3-27.
11. Загоруйко Н.Г. Прикладные методы анализа данных и знаний. Новосибирск: Изд-во Ин-та математики, 1999.
12. Загоруйко Н.Г. Когнитивный анализ данных. Новосибирск: ГЕО, 2013.