**НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ**

**УНИВЕРСИТЕТ**

**«МЭИ»**

Институт Автоматики и вычислительной техники

Кафедра Прикладной математики

**В Ы П У С К Н А Я Р А Б О Т А**

бакалавра Прикладной математики и информатики

**по направлению «Прикладная математика и информатика» (010500)**

Тема **Разработка системы прогнозирования биржевых котировок.**

Студент А-13-14 Пильненьский Н.Д. .

группа подпись фамилия, и., о.,

Научный   
руководитель доцент к.т.н Михайлов И.С. .

должность звание подпись фамилия, и., о.,

Консультант .

должность звание подпись фамилия, и., о.,

«Работа допущена к защите»

Зав. кафедрой д. т. н. Еремеев А. П. .

звание подпись фамилия, и., о.,

Дата\_\_\_\_\_\_\_\_\_\_\_\_\_

Москва 2018 г.

**АННОТАЦИЯ**

В данной работе разработана экспертная система определения тарифного плана сотовой связи, основанная на продукционной модели знаний и механизме нечеткой логики.

Экспертная система задает вопросы пользователю, направленные на определение его предпочтений в использовании сотовой связи, затем осуществляет подбор подходящего тарифного плана, основываясь на полученных от пользователя ответах. Языком разработки системы является C++.

Применение данной системы позволит не только эффективно подобрать подходящий тариф сотовой связи, но и сэкономить большое количество времени.

В данной работе разработана система прогнозирования биржевых котировок, основанная на машинном обучении, рекурентных нейронных сетях и механизме статистики.

**THE ANNOTATION**

In this work an expert system of determining the tariff plan of the mobile connection, based on a production model of knowledge and mechanism of fuzzy logic, has been developed.

Expert system asks the user to determine his preferences in using of mobile connection, and then select the optimum tariff plan, based on the users’ answers. The system is developed on C++ language.

This system allows not only finding a suitable tariff of mobile connection, but also saving a lot of time.

Оглавление

[Введение 5](#_Toc359245995)

[1. Анализ предметной области 7](#_Toc359245996)

[1.1. Основные понятия нечеткой логики 7](#_Toc359245997)

[1.2. Основные понятия модели знаний 8](#_Toc359245998)

[1.3. Основные понятия экспертной системы 12](#_Toc359245999)

[Выводы по главе 1 16](#_Toc359246000)

[2. Разработка модели представления знаний о тарифах сотовых операторов и предпочтениях пользователя 18](#_Toc359246001)

[2.1. Основные параметры тарифных планов сотовых компаний 18](#_Toc359246002)

[2.2. Описание параметров тарифных планов в виде лингвистических переменных 19](#_Toc359246003)

[2.2.1. Лингвистические переменные для тарифов, ориентированных на местную связь 19](#_Toc359246004)

[2.2.2. Лингвистические переменные для тарифов, ориентированных на междугороднюю связь 23](#_Toc359246005)

[2.2.3. Лингвистические переменные для тарифов, ориентированных на международную связь 27](#_Toc359246006)

[2.2.4. Лингвистические переменные для тарифов, ориентированных на мобильный интернет 29](#_Toc359246007)

[2.3. Преобразование условий тарифных планов в нечеткие правила 31](#_Toc359246008)

[Выводы по главе 2 34](#_Toc359246009)

[3. Разработка метода определения коэффициента покрытия на основе нечеткой логики 35](#_Toc359246010)

[3.1. Определение коэффициента покрытия 35](#_Toc359246011)

[3.2. Разработка метода определения коэффициента покрытия для определенной области 41](#_Toc359246012)

[Выводы по главе 3 43](#_Toc359246013)

[4. РАЗРАБОТКА ПРОГРАММНОЙ РЕАЛИЗАЦИИ ЭКСПЕРТНОЙ СИСТЕМЫ 44](#_Toc359246014)

[4.1. Разработка архитектуры программы 44](#_Toc359246015)

[4.2. Разработка базы данных для системы 45](#_Toc359246016)

[4.3. Разработка интерфейса системы 48](#_Toc359246017)

[4.4. Пример работы системы 53](#_Toc359246018)

[Выводы по главе 4 55](#_Toc359246019)

[ЗАКЛЮЧЕНИЕ 56](#_Toc359246020)

[БИБЛИОГРАФИЧЕСКИЙ СПИСОК 58](#_Toc359246021)

[ПРИЛОЖЕНИЯ 59](#_Toc359246022)

[Приложение 1. Исходный код некоторых модулей программы 59](#_Toc359246023)

# Введение

В настоящее время сотовая связь все сильнее проникает в нашу жизнь. Многие уже не могут представить себя без мобильного телефона в руке. В такой ситуации подобрать нужный тариф – это главная проблема, для решения которой требуется немало усилий и времени. Ведь правильный выбор подходящего провайдера и тарифного плана помогает сэкономить деньги и нервы.

На данный момент существует огромное количество различных тарифных планов, разобраться в хитросплетениях которых абоненту зачастую бывает очень сложно. Стоимость звонков зависит от многих показателей, таких как длительность разговоров, количество отправляемых сообщений, объема использования Интернета, направления вызовов и т.д.

Цель представленной работы – разработать экспертную систему определения тарифного плана мобильной связи, соответствующего предпочтениям пользователя.

Для достижения поставленной цели в работе необходимо решить следующие задачи:

1. Выполнить анализ принципов построения экспертных систем.
2. Исследовать модели знаний, используемые в экспертных системах.
3. Разработать модель для представления знаний в разрабатываемой экспертной системе.
4. Разработать набор правил для экспертной системы, позволяющих выявить тарифный план, соответствующий предпочтениям пользователя.
5. Разработать метод определения коэффициента покрытия на основе механизма нечёткой логики.
6. Выполнить реализацию экспертной системы.

В первой главе данной работы даются основные понятия нечеткой логики, модели знаний и экспертной системы. Во второй главе данной работы даётся детальное описание разработки модели представления знаний о тарифах сотовых операторов. В третьей главе описывается разработка метода определения коэффициента покрытия на основе нечеткой логики. Детали проектирования, описание интерфейса и пример работы системы приведены в четвертой главе. В заключении представлены основные результаты настоящей работы.

# 1. Анализ предметной области

## 1.1. Основные понятия нечеткой логики

Подход к формализации понятия нечеткого множества состоит в обобщении понятия принадлежности. В обычной теории множеств существует несколько способов задания множества. Одним из них является задание с помощью характеристической функции, определяемой следующим образом.   
Пусть U — так называемое универсальное множество, из элементов которого образованы все остальные множества, рассматриваемые в данном классе задач, например множество всех целых чисел, множество всех гладких функций и т.д. *Характеристическая функция* множества — это функция , значения которой указывают, является ли элементом множества A:

. (1.1)

С точки зрения характеристической функции, нечеткие множества есть естественное обобщение обычных множеств, когда мы отказываемся от бинарного характера этой функции и предполагаем, что она может принимать любые значения на отрезке [0,1]. В теории нечетких множеств характеристическая функция называется *функцией принадлежности*, а ее значение — степенью принадлежности элемента *x* нечеткому множеству A [6].

Более строго, *нечетким множеством A* называется совокупность пар

где — функция принадлежности, т.е. .

*Нечеткая переменная* характеризуется тройкой (a, X, A), где

* a - имя переменной,
* X - универсальное множество (область определения a),
* A - нечеткое множество на X, описывающее ограничение на значение нечеткой переменной a [15].

*Лингвистическая переменная* характеризуется пятеркой  
 *(X, T(X),U,G,M)*, где:

* *X* — название переменной;
* *T(X)* — терм-множество переменной X, т.е. множество названий лингвистических значений переменной X, причем каждое из таких значений является нечеткой переменной x\* со значениями из универсального множества U с базовой переменной u;
* *U* - универсальное множество с базовой переменной u;
* *G* — синтаксическое правило, порождающее названия x\* значений переменной X;
* *M* — семантическое правило, которое ставит в соответствие каждой нечеткой переменной x\* ее смысл M(x\*), т.е. нечеткое подмножество M(x\*) универсального множества U [15].

Лингвистическая переменная отличается от числовой переменной тем, что ее значениями являются не числа, а слова или предложения в естественном или формальном языке. Поскольку слова, в общем, менее точны, чем числа, понятие лингвистической переменной дает возможность приближенно описывать явления, которые настолько сложны, что не поддаются описанию в общепринятых количественных терминах. В частности, нечеткое множество, которое представляет собой ограничение, связанное со значениями лингвистической переменной, можно рассматривать как совокупную характеристику различных подклассов элементов универсального множества.

## 1.2. Основные понятия модели знаний

*Система искусственного интеллекта* – это система, оперирующая знаниями о проблемной области. Без базы знаний систем искусственного интеллекта не существует. Для формализации и представления знаний разрабатываются специальные модели представления знаний и языки для описания знаний, выделяются различные типы знаний.

Модели представления знаний относятся к прагматическому направлению (основано на предположении о том, что мыслительная деятельность человека – «черный ящик») в исследованиях по искусственному интеллекту. При этом в системах, основанных на знаниях, информационные структуры представляются в форме декларативных (описательных) знаний, а алгоритмы и эвристики – в форме процедурных знаний.

Признано, что системы, основанные на знаниях, целесообразно разрабатывать с выделением ряда подсистем. Основными из них считаются:

• подсистема представления декларативных знаний;

• подсистема представления процедурных знаний;

• подсистема манипулирования знаниями (механизм логического вывода).

Можно условно классифицировать знания в предметной области на понятийные, конструктивные, процедурные, фактографические знания и метазнания.

*Понятийные знания* - набор понятий, используемых при решении данной задачи, например, в фундаментальных науках и теоретических областях наук, т.е. это понятийный аппарат науки.

*Конструктивные знания* - наборы структур, подсистем системы и взаимодействий между их элементами, например, в технике.

*Процедурные знания* - методы, процедуры (алгоритмы) их реализации и идентификации, например, в прикладных науках.

*Фактографические*- количественные и качественные характеристики объектов и явлений, например, в экспериментальных науках.

*Метазнания* - знания о порядке и правилах применения знаний.

*Модели представления знаний* – это одно из важнейших направлений исследований в области искусственного интеллекта. На сегодняшний день разработано уже достаточное количество моделей. Каждая из них обладает своими плюсами и минусами, и поэтому для каждой конкретной задачи необходимо выбрать подходящую модель. От этого зависит не столько эффективность выполнения поставленной задачи, сколько возможность ее решения вообще.

Рассмотрим три наиболее часто используемые и популярные на сегодняшний день модели представления знаний: продукционные, сетевые, фреймовые.

*Сетевые модели* или семантические сети – как правило, это граф, отображающий смысл целостного образа. Узлы графа соответствуют понятиям и объектам, а дуги – отношениям между объектами.

*Фреймовые модели* – основывается на таком понятии как фрейм (англ. frame – рамка, каркас). Фрейм – структура данных для представления некоторого концептуального объекта. Информация, относящаяся к фрейму, содержится в составляющих его слотах. Слоты могут быть терминальными либо являться фреймами, образуя целую иерархическую сеть.

В данной работе для представления знания была выбрана продукционная модель, поэтому остановимся на ней более подробно.

*Продукционные модели* – модели, основанные на правилах, позволяют представить знание в виде предложений типа: «ЕСЛИ условие, ТО действие».

В общем случае продукционную модель можно представить в следующем виде:

i = <S; L; A \rightarrow B; Q>, где:

~S  — описание класса ситуаций;

~L  — условие, при котором продукция активизируется;

~A \rightarrow B  — ядро продукции;

Q — постусловие продукционного правила [14].

Системы обработки знаний, использующие продукционную модель, получили название «продукционных систем». В состав экспертных систем продукционного типа входят: база правил (знаний), рабочая память и интерпретатор правил (решатель), реализующий механизм логического вывода. Любое продукционное правило, содержащееся в базе знаний, состоит из двух частей: антецедента и консеквента. Антецедент представляет собой посылку правила (условную часть) и состоит из элементарных предложений, соединенных логическими связками «и», «или». Консеквент (заключение) включает одно или несколько предложений, которые выражают либо некоторый факт, либо указание на определенное действие, подлежащее исполнению. Продукционные правила принято записывать в виде антецедент-консеквент.

Любое правило состоит из одной или нескольких пар «атрибут-значение». В рабочей памяти систем, основанных на продукционных моделях, хранятся пары атрибут-значение, истинность которых установлена в процессе решения конкретной задачи к некоторому текущему моменту времени. Содержимое рабочей памяти изменяется в процессе решения задачи. Это происходит по мере срабатывания правил. Правило срабатывает, если при сопоставлении фактов, содержащихся в рабочей памяти, с антецедентом анализируемого правила имеет место совпадение, при этом заключение сработавшего правила заносится в рабочую память. Поэтому в процессе логического вывода объём фактов в рабочей памяти, как правило, увеличивается (уменьшаться он может в том случае, если действие какого-нибудь правила состоит в удалении фактов из рабочей памяти). В процессе логического вывода каждое правило из базы правил может сработать только один раз.

Существуют два типа продукционных систем – с «прямыми» и «обратными» выводами. Прямые выводы реализуют стратегию «от фактов к заключениям». При обратных выводах выдвигаются гипотезы вероятностных заключений, которые могут быть подтверждены или опровергнуты на основании фактов, поступающих в рабочую память. Существуют также системы с двунаправленными выводами.

Основные достоинства систем, основанных на продукционных моделях, связаны с простотой представления знаний и организации логического вывода. К недостаткам таких систем можно отнести следующее:

* отличие от структур знаний, свойственных человеку;
* неясность взаимных отношений правил;
* сложность оценки целостного образа знаний;
* низкая эффективность обработки знаний.

При разработке небольших систем (десятки правил) проявляются в основном положительные стороны продукционных моделей знаний, однако, при увеличении объёма знаний более заметными становятся слабые стороны.

## 1.3. Основные понятия экспертной системы

*Экспертная система (ЭС) -* система искусственного интеллекта, содержащая знания об определенной слабо структурированной и трудно формализуемой узкой предметной области и способная предлагать пользователю разумные решения и их объяснения. Экспертные системы играют роль эксперта и решают задачи диагностики, прогнозирования или планирования.

Рассмотрим примеры наиболее известных экспертных систем, с которых началось создание и развитие этого типа программных средств:

* *MYCIN* - это экспертная система, разработанная в начале 1970х годов в Стэндфордском университете. Система предназначена для диагностики и лечения медицинских инфекций. Исходя из представленных пациентом симптомов, система ставит диагноз и рекомендует курс соответствующего медикаментозного лечения.
* *PROSPECTOR* - экспертная система, применяемая при поиске месторождений полезных ископаемых. Система делает выводы на основе геологических анализов.
* *DENDRAL* - это старейшая экспертная система в мире. Экспертная система автоматизировала процесс определения химической структуры вещества и определяла строение органических молекул по химическим формулам и спектрографическим данным о химических связях в молекулах.

Экспертные системы бывают двух типов: статические и динамические. Статические экспертные системы не изменяются с течением времени, а динамические изменяются, приспосабливаясь к изменениям внешнего мира. Например, динамические экспертные системы могут использовать показания каких-либо датчиков или заимствовать дополнительную информацию из сети Internet.

Поскольку данная дипломная работа посвящена созданию экспертной системы статического типа, то на них мы остановимся подробнее. Рассмотрим структуру статической экспертной системы (рис.1.1).

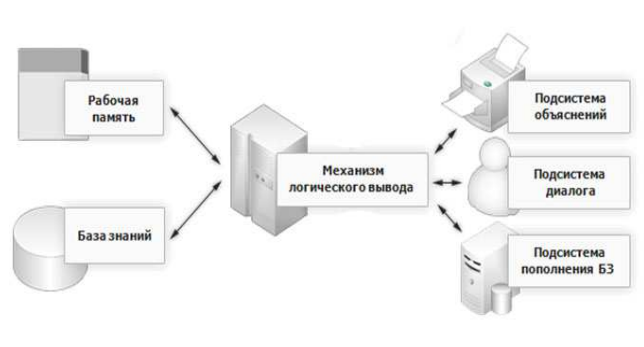


Рис.1.1. Структура экспертной системы.

Структура экспертной системы содержит следующие объекты:

• база знаний

• рабочая память (база данных)

• механизм логического вывода

• подсистема диалога

• подсистема приобретения и пополнения знаний

• подсистема объяснения

*База знаний* — это одно из ключевых понятий в области экспертных систем, означающее совокупность знаний (фактов и правил) о предметной области, обычно на некотором языке, близком к естественному. Под правилами в базе знаний понимаются правила вывода, описывающие отношения между фактами. Факты же имеют классическое значение и представляют собой знания в форме утверждений.

Помимо правил и фактов в базу знаний может входить некоторая процедурная часть, выполняющая расчетные, оптимизационные или другие нужные алгоритмы.

*Рабочая память* предназначена для временного хранения исходных данных и промежуточных фактов (полученных от пользователя) решаемой в текущий момент задачи. Как правило, рабочая память размещается в оперативной памяти ЭВМ.

*Механизм логического вывода* предназначен для получения новых фактов на основе сопоставления исходных данных из рабочей памяти и знаний из базы знаний. Механизм логического вывода во всей структуре экспертной системы занимает наиболее важное место, т.к. именно он моделирует ход рассуждений эксперта.

*Подсистема диалога* отвечает за организацию интерфейса для общения с пользователем в процессе решения задачи и получения результата. Именно этот элемент экспертной системы в дальнейшем будет называться пользовательским интерфейсом.

*Подсистема приобретения и пополнения знаний* автоматизирует процесс наполнения экспертной системы знаниями, осуществляемый пользователем-экспертом, и адаптации базы знаний системы к условиям ее функционирования. Адаптация экспертной системы к изменениям в предметной области реализуется путем замены правил или фактов в базе знаний.

*Подсистема объяснения* позволяет пользователю понять, каким образом система пришла к итоговому решению задачи или почему решение не было найдено. Это повышает доверие пользователя к полученному результату и упрощает разработчику тестирование программы. Возможность объяснять свои действия является одним из самых важных свойств экспертной системы, потому что еще больше приближает ее работу к работе эксперта в данной области.

Логико-лингвистические методы описания систем основаны на том, что поведение исследуемой системы описывается в естественном (или близком к естественному) языке в терминах лингвистических переменных. Входные и выходные параметры системы рассматриваются как лингвистические переменные, а качественное описание процесса задается совокупностью высказываний следующего вида:

L1 если A11 и/или A12 и/или ... и/или A1m, то B11 и/или ... и/или B1n,

L2 если A21 и/или A22 и/или ... и/или A2m, то B21 и/или ... и/или B2n,

.....................

Lk если Ak1 и/или Ak2 и/или ... и/или Akm, то Bk1 и/или ... и/или Bkn,

где Aij — нечеткие высказывания, определенные на значениях входных лингвистических переменных, а Bij - нечеткие высказывания, определенные на значениях выходных лингвистических переменных. Эта совокупность правил носит название *нечеткой базы знаний*.

Подобные вычисления составляют основу нечетких экспертных систем. Каждая *нечеткая экспертная система* использует нечеткие утверждения и правила.

В основе построения логико-лингвистических систем лежит композиционное правило вывода Заде.

Преимущество данной модели - в ее универсальности. Нам неважно, что именно на входе — конкретные числовые значения или некоторая неопределенность, описываемая нечетким множеством. Но за данную универсальность приходится расплачиваться сложностью системы — нам приходится работать в пространстве размерности. Поэтому этой общей моделью на практике пользуются довольно редко. Обычно же используют ее упрощенный вариант, называемый нечетким выводом. Он основывается на предположении, что все входные лингвистические переменные имеют известные — нечеткие высказывания, определенные на значениях выходных нам числовые значения (как и бывает довольно часто на практике). Также обычно не используют более одной выходной лингвистической переменной.

*Нечетким логическим выводом* (fuzzy logic inference) называется аппроксимация зависимости каждой выходной лингвистической переменной от входных лингвистических переменных и получение заключения в виде нечеткого множества, соответствующего текущим значениям входов, с использованием нечеткой базы знаний и нечетких операций.

## Выводы по главе 1

* + 1. В данной главе были рассмотрены основные понятия нечеткой логики, модели знаний и экспертной системы.
    2. Для данной экспертной системы была выбрана продукционная модель построения базы знаний, так как она является наиболее наглядным средством обработки знаний, представленных в виде набора условий. Продукционная модель привлекает своей наглядностью, высокой модульностью, легкостью внесения дополнений и изменений и простотой процедуры вывода.
    3. Для создания алгоритма определения подходящего тарифного плана был выбран механизм нечёткой логики, так как он обеспечивает представление о тарифах сотовых компаний в виде лингвистических переменных, значения которых может задавать пользователь на ограниченном естественном языке. Механизм нечёткой логики позволяет обрабатывать большое количество информации о сотовых вышках и объединении их площадей покрытия, а также получать результат в зависимости от предпочтения пользователя по качеству приёма сигнала.

# 2. Разработка модели представления знаний о тарифах сотовых операторов и предпочтениях пользователя

## 2.1. Основные параметры тарифных планов сотовых компаний

В данной работе в качестве примера была рассмотрена Калининградская область и соответствующие ей тарифные планы и цены на них.

Большинство тарифных планов ориентированы на конкретный вид связи, иногда на несколько видов. Будем рассматривать следующие виды связи: местная связь, междугородняя связь, международная связь, мобильный интернет. Каждый тариф, ориентированный на определенный вид связи, характеризуется своими параметрами. Выделим основные параметры.

Параметры тарифов, ориентированных на местную связь:

* стоимость первой минуты местного вызова на своего оператора;
* стоимость со 2-й по 10-ю минуту местного вызова на своего оператора;
* стоимость с 11-й минуты местного вызова на своего оператора;
* стоимость местной минуты на других операторов;
* стоимость местной SMS;

Параметры тарифов, ориентированных на междугороднюю связь:

* стоимость междугородней первой минуты на своего оператора;
* стоимость со 2-й по 10-ю междугороднюю минуту на своего оператора;
* стоимость с 11-й междугородней минуты на своего оператора;
* стоимость междугородней минуты на других операторов;
* стоимость междугородней SMS;

Параметры тарифов, ориентированных на международную связь:

* стоимость международной первой минуты;
* стоимость со 2-й по 10-ю международную минуту;
* стоимость с 11-й международной минуты;
* стоимость международной SMS;
* стоимость международной минуты на других операторов;

Параметры тарифов, ориентированных на мобильный интернет:

* стоимость MMS;
* стоимость GPRS трафика.

## 2.2. Описание параметров тарифных планов в виде лингвистических переменных

## 2.2.1. Лингвистические переменные для тарифов, ориентированных на местную связь

**,** где

* *X*: Стоимость местной первой минуты .
* *T(X):* T(Стоимость местной первой минуты ) = {«Высокая», «Средняя», «Низкая»}. Функции принадлежности термов [15]:

,

,

.

Графики функции принадлежности термов имеют следующий вид (рис. 2.1):

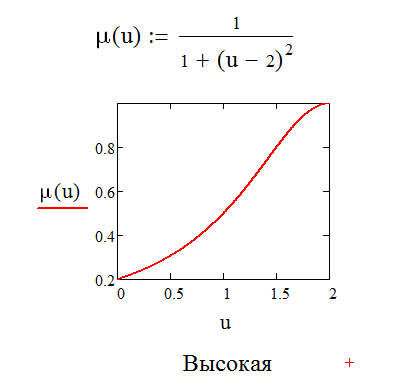
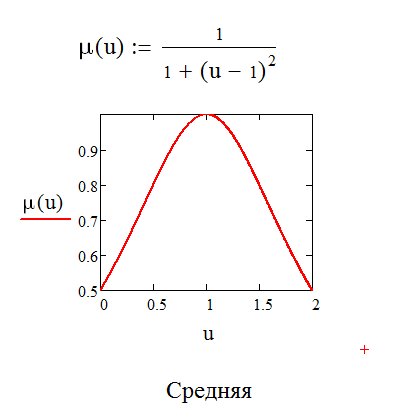




Рис.2.1. Графики функций принадлежности термов переменной «Стоимость местной первой минуты на своего оператора»

* *U* = [0; 2.
* *G*: "не", "очень".
* *М*: если терм t имеет функцию принадлежности , то новые термы будут иметь следующие функции принадлежности:

|  |  |
| --- | --- |
| Квантификатор | Функция принадлежности |
| очень t |  |
| не t | 1 - |

**,** где

* *X*: **.**
* *T(X):* T() = {«Высокая», «Средняя», «Низкая»}. Функции принадлежности термов:

,

,

.

* *U* = [0; 2.
* *G:* "не", "очень".
* *М*: если терм t имеет функцию принадлежности , то новые термы будут иметь следующие функции принадлежности, заданные в таблице:

|  |  |
| --- | --- |
| Квантификатор | Функция принадлежности |
| очень t |  |
| не t | 1 - |

, где

* *X*: Стоимость с 11-й местной минуты .
* *T(X):* T(Стоимость с 11-й местной минуты ) = {«Высокая», «Средняя», «Низкая»}. Функции принадлежности термов:

,

,

.

* *U* = [0; 2.
* *G*: "не", "очень".
* *М*: если терм t имеет функцию принадлежности , то новые термы будут иметь следующие функции принадлежности, заданные в таблице:

|  |  |
| --- | --- |
| Квантификатор | Функция принадлежности |
| очень t |  |
| не t | 1 - |

, где

* *X*: Стоимость местной первой SMS.
* *T(X)*: T(Стоимость местной SMS) = {«Высокая», «Средняя», «Низкая»}. Функции принадлежности термов:

,

,

.

* U = [0; 2.
* *G*: "не", "очень".
* *М*: если терм t имеет функцию принадлежности , то новые термы будут иметь следующие функции принадлежности, заданные в таблице:

|  |  |
| --- | --- |
| Квантификатор | Функция принадлежности |
| очень t |  |
| не t | 1 - |

**,** где

* *X*: Стоимость местной минуты .
* *T(X)*: T(Стоимость местной минуты ) = {«Высокая», «Средняя», «Низкая»}. Функции принадлежности термов:

,

,

.

* *U* = [0; 2.
* *G*: "не", "очень".
* *М*: если терм t имеет функцию принадлежности , то новые термы будут иметь следующие функции принадлежности, заданные в таблице:

|  |  |
| --- | --- |
| Квантификатор | Функция принадлежности |
| очень t |  |
| не t | 1 - |

## 2.2.2. Лингвистические переменные для тарифов, ориентированных на междугороднюю связь

**,** где

* *X*: Стоимость междугородней первой минуты .
* *T(X):* T(Стоимость междугородней первой минуты ) = {«Высокая», «Средняя», «Низкая»}. Функции принадлежности термов [15]:

,

,

.

* *U* = [0; 13.
* *G*: "не", "очень".
* *М:* если терм t имеет функцию принадлежности , то новые термы будут иметь следующие функции принадлежности, заданные в таблице:

|  |  |
| --- | --- |
| Квантификатор | Функция принадлежности |
| очень t |  |
| не t | 1 - |

**,** где

* *X*: **.**
* *T(X):* T() = {«Высокая», «Средняя», «Низкая»}. Функции принадлежности термов:

,

,

.

* *U* = [0; 13.
* *G*: "не", "очень".
* *М*: если терм t имеет функцию принадлежности , то новые термы будут иметь следующие функции принадлежности, заданные в таблице:

|  |  |
| --- | --- |
| Квантификатор | Функция принадлежности |
| очень t |  |
| не t | 1 - |

, где

* *X*: Стоимость с 11-й минуты .
* *T(X):* T(Стоимость с 11-й минуты) = {«Высокая», «Средняя», «Низкая»}. Функции принадлежности термов:

,

,

.

* *U* = [0; 13.
* *G*: "не", "очень".
* *М:* если терм t имеет функцию принадлежности , то новые термы будут иметь следующие функции принадлежности, заданные в таблице:

|  |  |
| --- | --- |
| Квантификатор | Функция принадлежности |
| очень t |  |
| не t | 1 - |

, где

* *X*: Стоимость SMS.
* T(X): T(Стоимость SMS) = {«Высокая», «Средняя», «Низкая»}. Функции принадлежности термов:

,

,

.

* *U* = [0; 3.
* *G*: "не", "очень".
* *М*: если терм t имеет функцию принадлежности , то новые термы будут иметь следующие функции принадлежности, заданные в таблице:

|  |  |
| --- | --- |
| Квантификатор | Функция принадлежности |
| очень t |  |
| не t | 1 - |

**,** где

* *X*: Стоимость междугородней минуты .
* *T(X)*: T(Стоимость междугородней минуты ) = {«Высокая», «Средняя», «Низкая»}. Функции принадлежности термов:

,

,

.

* *U* = [0; 13.
* *G*: "не", "очень".
* *М*: если терм t имеет функцию принадлежности , то новые термы будут иметь следующие функции принадлежности, заданные в таблице:

|  |  |
| --- | --- |
| Квантификатор | Функция принадлежности |
| очень t |  |
| не t | 1 - |

## 2.2.3. Лингвистические переменные для тарифов, ориентированных на международную связь

**,** где

* *X*: Стоимость первой минуты.
* *T(X):* T(Стоимость первой минуты) = {«Высокая», «Средняя», «Низкая»}. Функции принадлежности термов:

,

,

.

* *U* = [0; 70.
* *G*: "не", "очень".
* *М*: если терм t имеет функцию принадлежности , то новые термы будут иметь следующие функции принадлежности, заданные в таблице:

|  |  |
| --- | --- |
| Квантификатор | Функция принадлежности |
| очень t |  |
| не t | 1 - |

**,** где

* *X*: **.**
* *T(X)*: T() = {«Высокая», «Средняя», «Низкая»}. Функции принадлежности термов:

,

,

.

* *U* = [0; 70.
* *G:* "не", "очень".
* *М*: если терм t имеет функцию принадлежности , то новые термы будут иметь следующие функции принадлежности, заданные в таблице:

|  |  |
| --- | --- |
| Квантификатор | Функция принадлежности |
| очень t |  |
| не t | 1 - |

, где

* *X*: Стоимость с 11-й минуты.
* *T(X):* T(Стоимость с 11-й минуты) = {«Высокая», «Средняя», «Низкая»}. Функции принадлежности термов:

,

,

.

* *U* = [0; 70.
* *G:* "не", "очень".
* *М*: если терм t имеет функцию принадлежности , то новые термы будут иметь следующие функции принадлежности, заданные в таблице:

|  |  |
| --- | --- |
| Квантификатор | Функция принадлежности |
| очень t |  |
| не t | 1 - |

, где

* *X*: Стоимость SMS.
* *T(X):* T(Стоимость SMS) = {«Высокая», «Средняя», «Низкая»}. Функции принадлежности термов:

,

,

.

* *U* = [0; 7.
* *G*: "не", "очень".
* *М*: если терм t имеет функцию принадлежности , то новые термы будут иметь следующие функции принадлежности, заданные в таблице:

|  |  |
| --- | --- |
| **Квантификатор** | **Функция принадлежности** |
| очень t |  |
| не t | 1 - |

## 2.2.4. Лингвистические переменные для тарифов, ориентированных на мобильный интернет

, где

* *X*: Стоимость .
* *T(X)*: T(Стоимость ) = {«Высокая», «Средняя», «Низкая»}. Функции принадлежности термов:

,

,

.

* *U* = [6; 10.
* *G*: "не", "очень".
* *М*: если терм t имеет функцию принадлежности , то новые термы будут иметь следующие функции принадлежности, заданные в таблице:

|  |  |
| --- | --- |
| Квантификатор | Функция принадлежности |
| очень t |  |
| не t | 1 - |

, где

* *X*: .
* *T(X)*: T(Стоимость ) = {«Высокая», «Средняя», «Низкая»}. Функции принадлежности термов:

,

,

.

* *U* = [0; 10.
* *G*: "не", "очень".
* *М*: если терм t имеет функцию принадлежности , то новые термы будут иметь следующие функции принадлежности, заданные в таблице:

|  |  |
| --- | --- |
| Квантификатор | Функция принадлежности |
| очень t |  |
| не t | 1 - |

## 2.3. Преобразование условий тарифных планов в нечеткие правила

Для преобразования условий тарифного плана в нечеткое правило необходимо для каждого параметра тарифа вычислить значения функций принадлежности термов соответствующей ему лингвистической переменной и найти наибольшее среди них.

Приведем пример такого преобразования для тарифного плана «Ноль сомнений» оператора Билайн. Рассматриваемый тарифный план ориентирован на местную связь. Приведем значения его параметров:

* местные вызовы на номера оператора Билайн:  
  1-я минута - разговора 0,60 руб.,  
  со 2-й по 10-ю минуту разговора - 0 руб.,  
  с 11-й минуты разговора - 0,60 руб.;
* стоимость вызова на местные номера других операторов - 1,20 руб.;
* передача SMS на местные номера - 1,20 руб.;

Вычислим значения функций принадлежности термов для каждого параметра тарифа.

;

;

;

Максимальное значение среди найденных равно , поэтому для тарифного плана «Ноль сомнений» оператора Билайн: «Стоимость местной первой минуты на своего оператора» = «Средняя».

;

;

;

Максимальное значение среди найденных равно 1, поэтому для тарифного плана «Ноль сомнений» оператора Билайн: «Стоимость со 2-й по 10-ю местную минуту на своего оператора» = «Низкая».

Аналогично, «Стоимость со 11-й местной минуты на своего оператора» = «Средняя».

;

;

;

Максимальное значение среди найденных равно , поэтому для тарифного плана «Ноль сомнений» оператора Билайн: «Стоимость местной минуты на других операторов» = «Средняя».

;

;

;

Максимальное значение среди найденных равно , поэтому для тарифного плана «Ноль сомнений» оператора Билайн: «Стоимость местной SMS» = «Средняя».

В итоге получаем нечеткие условия.

Тариф «Ноль сомнений» (Билайн): «Стоимость местной первой минуты на своего оператора» = «Средняя», «Стоимость со 2-й по 10-ю местную минуту на своего оператора» = «Низкая», «Стоимость c 11-й местной минуты на своего оператора» = «Средняя», «Стоимость местной минуты на других операторов» = «Средняя», «Стоимость местной SMS» = «Средняя».

Теперь нечеткое описание тарифа преобразуем в нечеткое правило для экспертной системы.

ЕСЛИ

«Стоимость местной первой минуты на своего оператора» = «Средняя»&«Стоимость со 2-й по 10-ю местную минуту на своего оператора» = «Низкая» &«Стоимость c 11-й местной минуты на своего оператора» = «Средняя» &«Стоимость местной минуты на других операторов» = «Средняя»&«Стоимость местной SMS» = «Средняя»

ТО

Тариф «Ноль сомнений» (Билайн).

Аналогичные преобразования необходимо выполнить для каждого тарифного плана. Так как данная процедура для большего количества тарифных планов займет огромное количество времени, то реализацию необходимых преобразований будет выполнять программа.

**Метод получения нечетких правил для экспертной системы.**

Входными параметрами метода являются условия тарифных планов.

* 1. Для параметра тарифного плана вычислить значения функций принадлежности термов соответствующей данному параметру лингвистической переменной.
  2. Найти максимальное среди полученных значений.
  3. Присвоить лингвистической переменной, которой соответствует рассматриваемый параметр тарифа, значение терма, соответствующее значению найденному в п.2.
  4. Проделать п.1-п.3 для всех рассматриваемых параметров тарифного плана.
  5. Создать нечеткое правило, объединяя полученные в п.4 пары «лингвистическая переменная – значение» и используя связки «ЕСЛИ» и «ТО».
  6. Выполнить п.5 для каждого тарифного плана.

В результате применения данного метода будет получен набор правил для разрабатываемой экспертной системы.

## Выводы по главе 2

1. Выполнен анализ предметной области тарифных планов сотовых компаний.

2. Выделены и сгруппированы основные параметры тарифных планов.

3. Разработан набор лингвистических переменных, характеризующих исходную информацию о тарифных планах.

4. Разработан метод получения набора правил для разрабатываемой экспертной системы.

# 3. Разработка метода определения коэффициента покрытия на основе нечеткой логики

## 3.1. Определение коэффициента покрытия

Для определения оператора, предоставляющего наилучшее качество покрытия сети 3G для конкретной области, необходимо вычислить коэффициент покрытия 3G сигнала, который зададим следующей формулой:

, (3.1)

где – площадь части рассматриваемой области, в которой уверенный прием сигнала 3G;

- площадь части рассматриваемой области, в которой средний прием сигнала 3G;

– площадь всей области.

В данной работе в качестве примера рассматривается Калининградская область.

Для вычисления коэффициента покрытия необходимы координаты 3G вышек и координаты границы рассматриваемой области.

Для нахождения координат точек, лежащих на границе области, были использованы карты Google (рис.3.1) и приложение Google earth (рис.3.2).

Пример географических координат граничных точек Калининградской области приведен в таблице 3.1.

Таблица 3.1.

|  |  |
| --- | --- |
| Северная широта | Восточная долгота |
| 54.328034 | 21.356953 |
| 54.335278 | 21.997487 |
| 54.368954 | 22.778452 |
| 54.579602 | 22.673133 |
| 54.830482 | 22.887180 |
| 55.061519 | 22.576795 |
| 55.105226 | 21.958288 |
| 55.214875 | 21.574494 |
| 55.285817 | 21.317734 |
| 55.264795 | 20.897377 |
| 55.132552 | 20.769133 |
| 55.030765 | 20.518596 |
| 54.959404 | 20.471483 |
| 54.950518 | 20.067603 |
| 54.816923 | 19.986237 |
| 54.736299 | 19.990382 |

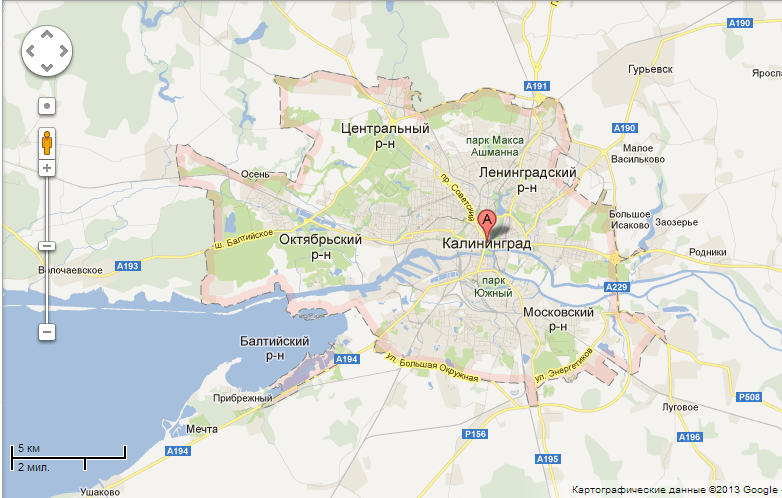


Рис.3.1. Карта Google Калининградской области



Рис.3.1. Карта Калининградской области в приложении Google earth

Для нахождения координат 3G вышек Калининградской области было использовано приложение Google Earth, куда были загружены карты покрытия подвижной радиотелефонной связи с официального сайта Роскомнадзора (Федеральной службы по надзору в сфере связи, информационных технологий и массовых коммуникаций). Эти карты совмещались с картами покрытия 3G с официальных сайтов операторов МТС (рис.3.3, рис.3.4), Билайн (рис.3.5, рис.3.6) и Мегафон (рис.3.7, рис.3.8).

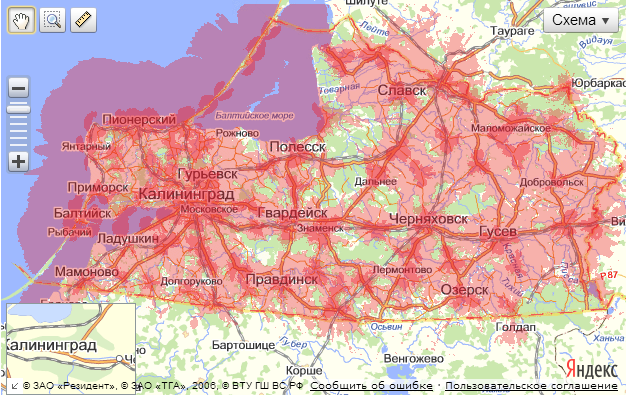


Рис 3.3. Карта покрытия Калининградской области сигналом 3G с официального сайта оператора МТС



Рис 3.4. Карта покрытия Калининградской области сигналом 3G оператора МТС в приложении Google Earth

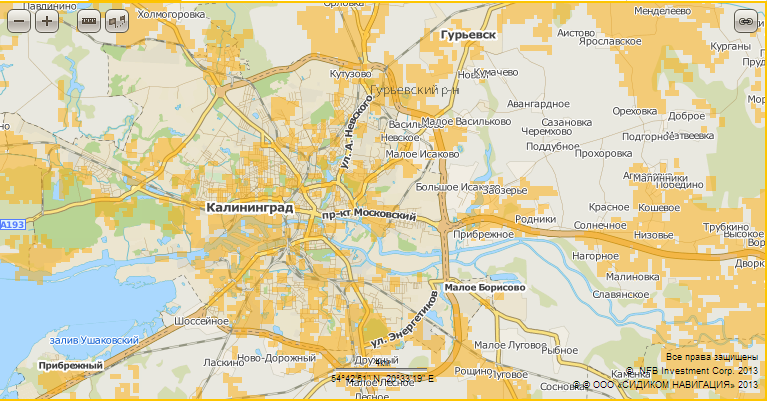


Рис 3.5. Карта покрытия Калининградской области сигналом 3G с официального сайта оператора Билайн



Рис 3.6. Карта покрытия Калининградской области сигналом 3G оператора Билайн в приложении Google Earth



Рис 3.7. Карта покрытия Калининградской области сигналом 3G с официального сайта оператора Мегафон

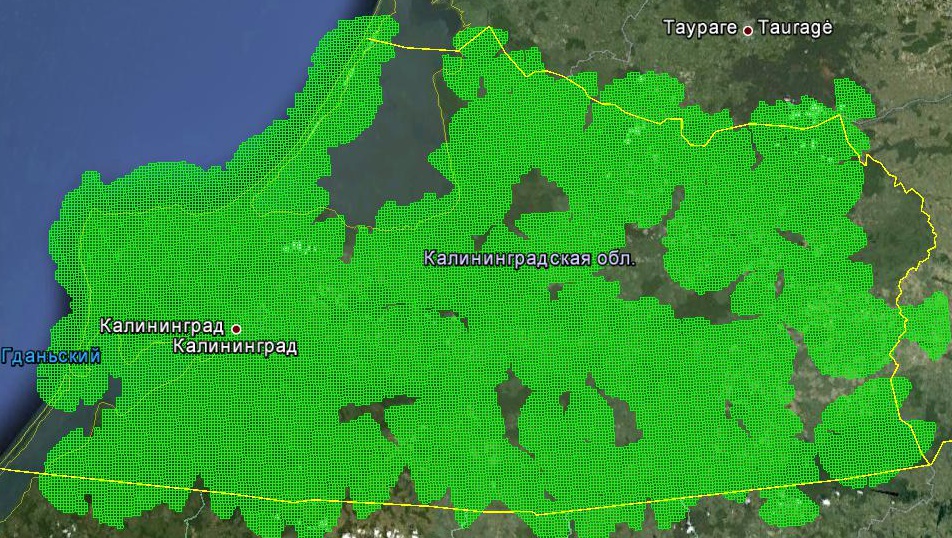


Рис 3.7. Карта покрытия Калининградской области сигналом 3G оператора Мегафон в приложении Google Earth

В Калининградской области было выделено 73 вышки 3G сигнала оператора МТС, 74 вышки – оператора Мегафон, 81 вышка – оператора Билайн.

Все необходимые данные для нахождения коэффициента покрытия 3 G сигнала найдены. Для того чтобы узнать величину площадей областей, в которых уверенный, средний и неуверенный прием 3G сигнала, необходимо определить уровень сигнала в каждой точке области.

## 3.2. Разработка метода определения коэффициента покрытия для определенной области

Вводим лингвистическую переменную «Прием 3G сигнала в точке области».

, где

* *X*: Прием 3G сигнала в точке области.
* *T(X)*: T(Прием 3G сигнала в точке области) = {«Уверенный прием», «Средний прием», «Неуверенный прием»}. Функции принадлежности термов:

,

,

,

где (, – координаты 3G вышки, (x,y) – координаты рассматриваемой точки.

* Калининградская область располагается в прямоугольнике [54.328; 55.286], следовательно, минимальное расстояние между рассматриваемой точкой и 3G вышкой равняется 0, а максимальное – 3.243 (максимальное расстояние между границами прямоугольника в градусах). Получаем универсальное множество U = [0; 3.243, в котором переменная u равна расстоянию в градусах от координаты рассматриваемой точки до координаты вышки.
* *G*: Использование квантификаторов "Очень", "Более-менее" позволяет создавать новые термы, например, «Очень уверенный прием».
* *М*: если терм t имеет функцию принадлежности , то новые термы будут иметь следующие функции принадлежности, заданные в таблице:

|  |  |
| --- | --- |
| Квантификатор | Функция принадлежности |
| очень t |  |
| более-менее t |  |

**Метод вычисления коэффициента покрытия области 3G сигналом определенного оператора.**

Входными параметрами метода являются: координаты границ рассматриваемой области, координаты 3G вышек.

* + 1. Для точки области найти ближайшую 3G вышку.
    2. Используя координаты найденной в п.1 вышки вычислить значения функций принадлежности всех термов лингвистической переменной «Прием 3G сигнала в точке области».
    3. Найти среди полученных в п.2 значений максимальное, определяя тем самым уровень приема 3G сигнала в рассматриваемой точке.
    4. Выполнить п.1-п.3 для всех точек области.
    5. Подсчитать количество точек с уверенным и средним приемом сигнала, так же подсчитать общее количество рассматриваемых точек.
    6. С помощью формулы (3.1) вычислить коэффициент покрытия, где вместо будет количество точек с уверенным приемом сигнала, вместо будет количество точек со средним приемом сигнала и вместо будет общее количество рассматриваемых точек.

В результате применения данного метода будет получен коэффициент покрытия рассматриваемой области 3G сигналом для определенного оператора.

Если пользователю важно иметь постоянно уверенный прием 3G сигнала, то поиск тарифа будет вестись среди тарифов оператора с наибольшим коэффициентом покрытия области 3G сигналом.

## Выводы по главе 3

1. Выполнен анализ предметной области покрытия 3G сигналом для рассматриваемых сотовых операторов.
2. Разработана лингвистическая переменная, характеризующая уровень приема 3G сигнала.
3. Разработан метод вычисления коэффициента покрытия области 3G сигналом.

# 4. РАЗРАБОТКА ПРОГРАММНОЙ РЕАЛИЗАЦИИ ЭКСПЕРТНОЙ СИСТЕМЫ

## 4.1. Разработка архитектуры программы

Начальным этапом проектирования экспертной системы является разработка ее архитектуры. Архитектура разрабатываемой системы включает в себя следующие компоненты (рис.4.1):

* интерфейс пользователя;
* решатель (механизм логического вывода);
* рабочая память;
* подсистема приобретения знаний;
* база знаний;
* подсистема объяснений;

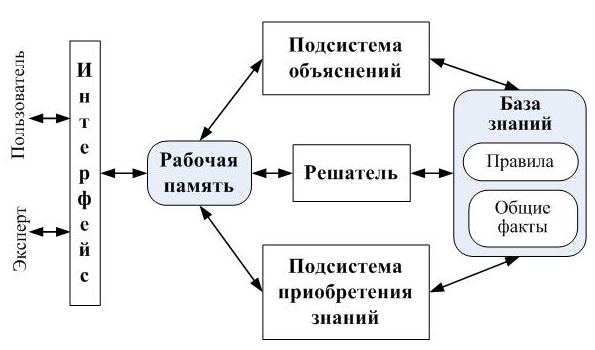


Рис 4.1. Архитектура разрабатываемой экспертной системы

*Интерфейс пользователя* выполняет следующие задачи:

* получение данных от пользователя (ответы на вопросы);
* выдача результата решения задачи.

*Рабочая память* предназначена для временного хранения исходных данных и промежуточных фактов (полученных от пользователя ответов на вопросы) решаемой в текущий момент задачи.

*Решатель (механизм логического вывода)* предназначен для получения новых фактов на основе сопоставления исходных данных из рабочей памяти и знаний из базы знаний.

*Подсистема приобретения и пополнения знаний* автоматизирует процесс наполнения экспертной системы знаниями, осуществляемый экспертом, также отвечает за адаптацию экспертной системы к изменениям в предметной области.

*Подсистема объяснения* позволяет пользователю понять, каким образом система пришла к итоговому решению задачи.

*База знаний* предназначена для хранения долгосрочных данных, описывающих рассматриваемую предметную область (а не текущих данных), и правил, описывающих целесообразные преобразования данных этой области.

## 4.2. Разработка базы данных для системы

Разработаем логическую и физическую модель базы данных разрабатываемой системы, которая будет содержать следующую информацию (рис.4.3, рис.4.4):

* операторы и условия тарифных планов;
* области и координаты их границ;
* координаты 3G вышек;
* нечеткие условия тарифных планов

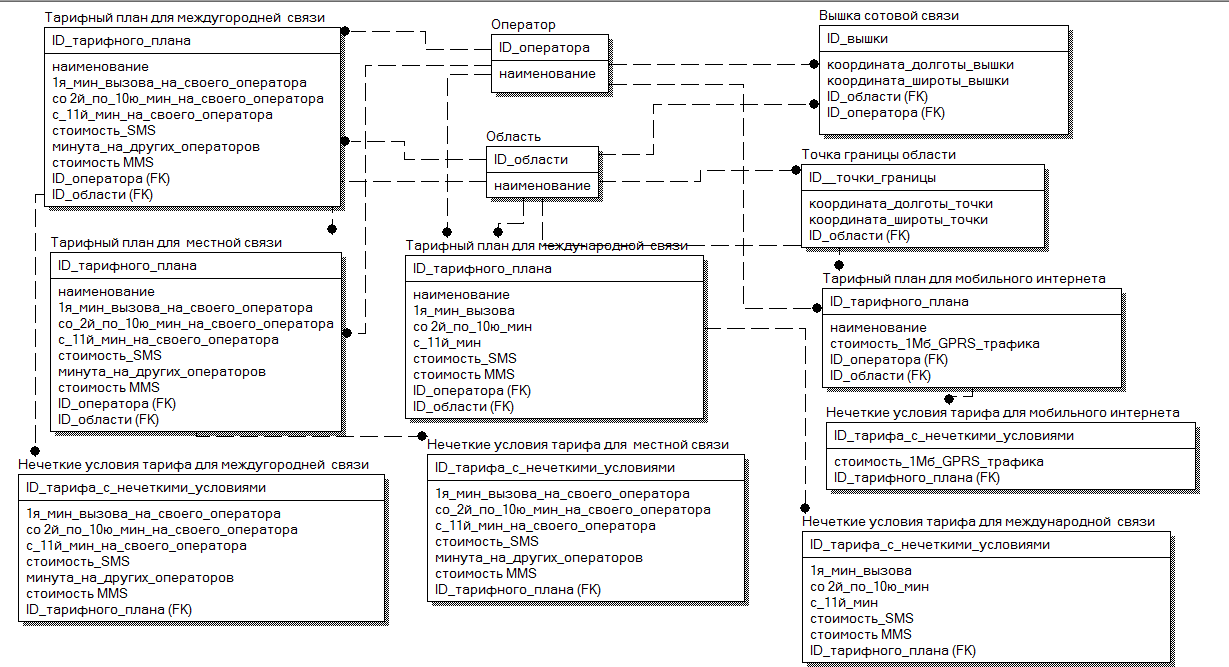


Рис 4.3. Логическая модель базы данных для разрабатываемой экспертной системы

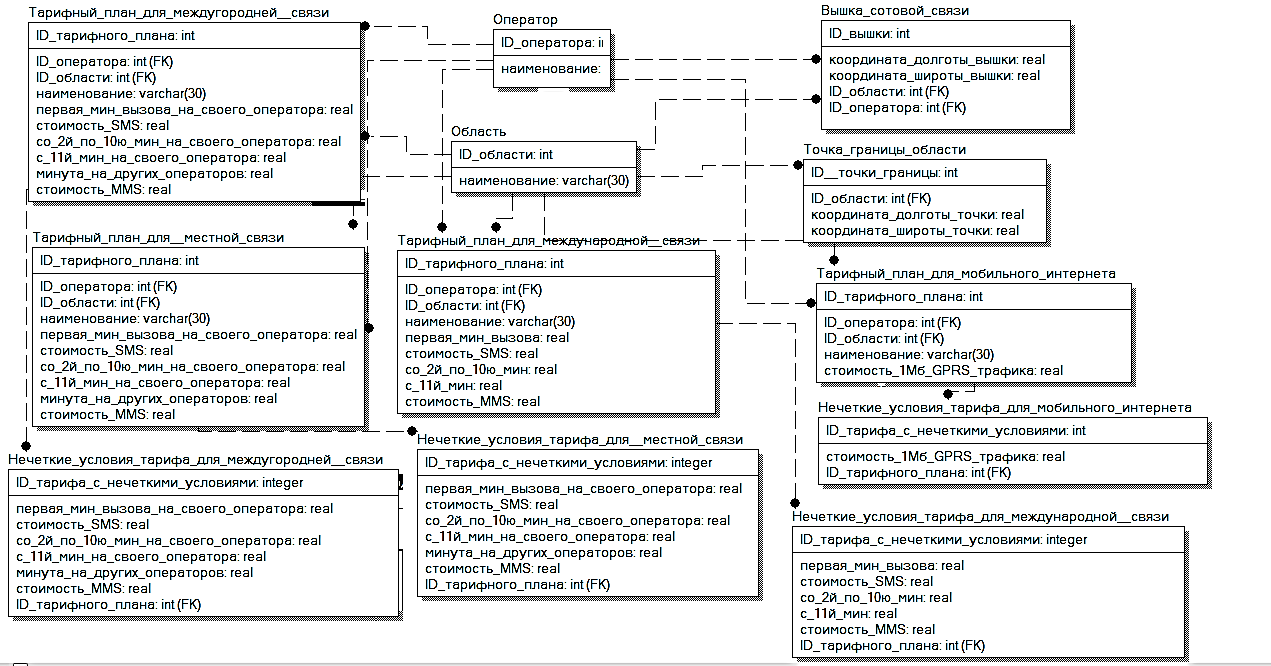


Рис 4.3. Физическая модель базы данных для разрабатываемой экспертной системы

Таблица «Тарифный план для междугородней связи» содержит условия тарифных планов, ориентированных на междугороднюю связь.

Таблица «Тарифный план для местной связи» содержит условия тарифных планов, ориентированных на местную связь.

Таблица «Тарифный план для международной связи» содержит условия тарифных планов, ориентированных на международную связь.

Таблица «Тарифный план для мобильного интернета» содержит условия тарифных планов, ориентированных на мобильный интернет.

Таблица «Область» содержит названия областей, с которыми работает система.

Таблица «Оператор» содержит названия операторов, тарифные планы которых рассматриваются системой.

Таблица «Вышка сотовой связи» содержит географические координаты 3G вышек.

Таблица «Нечеткие условия тарифа для междугородней связи» содержит нечеткие условия тарифных планов, ориентированных на междугороднюю связь и содержащихся в таблице «Тарифный план для междугородней связи».

Таблица «Нечеткие условия тарифа для международной связи» содержит нечеткие условия тарифных планов, ориентированных на международную связь и содержащихся в таблице «Тарифный план для международной связи».

Таблица «Нечеткие условия тарифа для местной связи» содержит нечеткие условия тарифных планов, ориентированных на местную связь и содержащихся в таблице «Тарифный план для местной связи».

Таблица «Нечеткие условия тарифа для мобильного интернета» содержит нечеткие условия тарифных планов, ориентированных на мобильный интернет и содержащихся в таблице «Тарифный план для мобильного интернета».

## 4.3. Разработка интерфейса системы

Программная реализация разрабатываемой экспертной системы была выполнена в среде Qt Creator на языке С++ с использованием библиотеки Qt. Данная среда была выбрана по причине следующих удобств:

* простота создания интерфейса;
* удобная работа с базами данных;
* наличие в библиотеке Qt огромного количества функций для работы с такими объектами как многоугольники, точки и прямые (в данной работе некоторые функции были использованы для определения принадлежности точки конкретной области).

Разработаем интерфейс, обеспечивающий диалог пользователя и эксперта с системой. Интерфейс должен быть прост в восприятии и должен отражать все возможности системы.

При запуске программы в ее главном окне расположено меню с выбором режима пользователя или разработчика (рис.4.4).

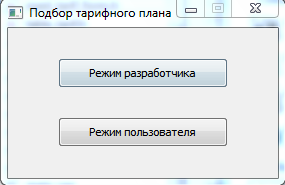


Рис 4.4. Главное окно программы

При выборе режима пользователя появляется окно с выбором вида связи, на который должен быть ориентирован тариф (рис.4.5).

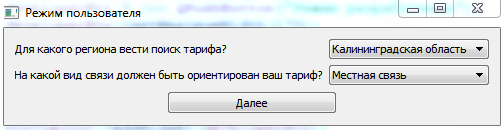


Рис 4.5. Окно с выбором вида связи

При нажатии кнопки «Далее» появляется окно с выбором параметров, соответствующих выбранному ранее виду связи (рис.4.6).

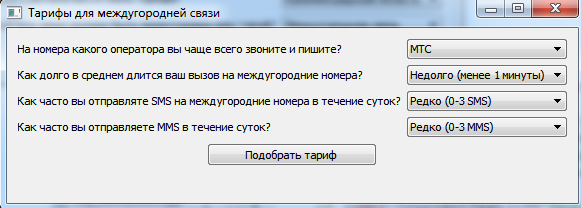


Рис 4.6. Окно с выбором параметров междугородней связи

При нажатии кнопки «Подобрать тариф» система ведет поиск подходящего тарифного плана по заданным параметрам и выводит результат (рис.4.7).

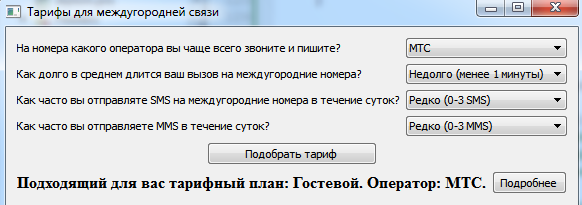


Рис 4.7. Вывод найденного тарифного плана

После нажатия кнопки «Подробнее» появляется окно с описанием условий найденного тарифного плана (рис.4.8).

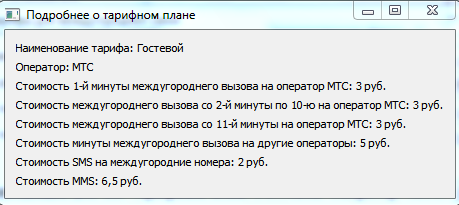


Рис 4.8. Условия найденного тарифного плана

При выборе вида связи «Мобильный интернет» появляется следующее окно (рис. 4.9):

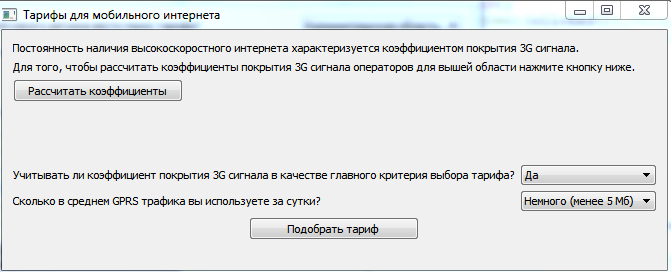


Рис.4.9. Окно с выбором параметров тарифов для мобильного интернета

При нажатии на кнопку «Рассчитать коэффициенты» происходит расчёт коэффициентов покрытия 3G сигнала для выбранной области, результаты выводятся на экран (рис.4.10).

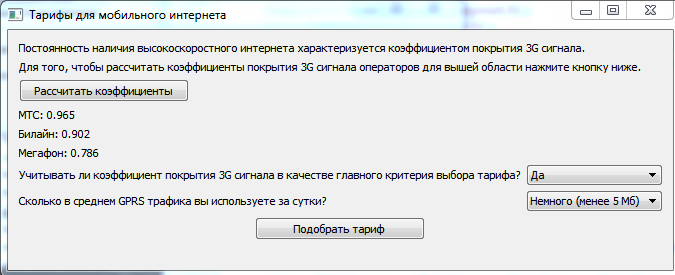


Рис. 4.10. Вывод на экран коэффициентов покрытия 3G сигналом

Если изначально был выбран режим разработчика, то появляется окно с выбором данных для редактирования (рис.4.11).

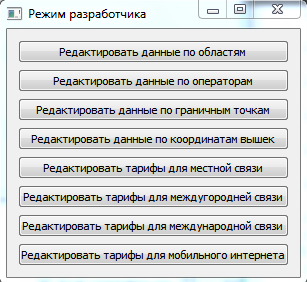


Рис 4.11. Окно с выбором данных для редактирования

После выбора данных для редактирования появляется окно редактора данных (рис.4.12).

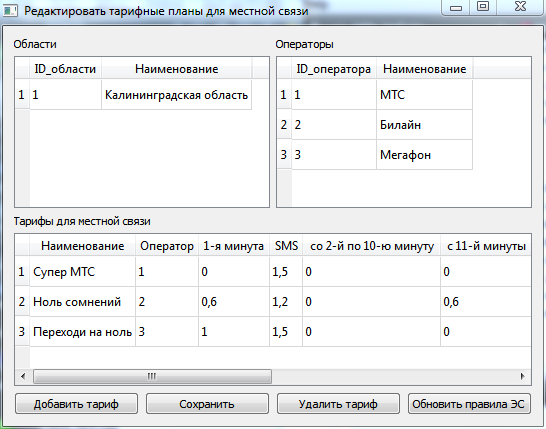


Рис 4.12. Окно редактора данных

## 4.4. Пример работы системы

Предположим, пользователю, живущему в Калининградской области, необходим тарифный план для общения с абонентами других городов. Тогда в открывшемся перед ним окне приложения ему нужно выбрать регион и соответствующий его предпочтениям вид связи, а именно: регион – Калининградская область, вид связи – междугородняя связь (рис. 4.13).

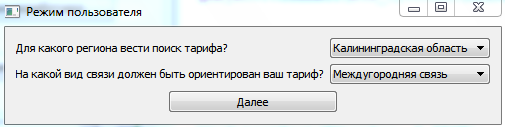


Рис 4.13. Окно с выбором вида связи

Нажав «Далее» перед пользователем открывается окно с выбором параметров выбранного вида связи. Предположим, пользователь чаще всего звонит на номера оператора МТС и средняя длительность его разговора с абонентами других городов очень мала. Пусть также пользователь редко пользуется такими услугами как MMS и отправка SMS на междугородние номера. Тогда соответствующие параметры ему необходимо выбрать в открывшемся окне (рис.4.14).

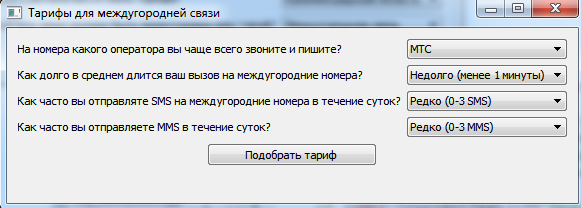


Рис 4.14. Окно с выбором параметров, соответствующих предпочтениям пользователя, для междугородней связи

При нажатии кнопки «Подобрать тариф» система выводит следующий результат (рис.4.15).

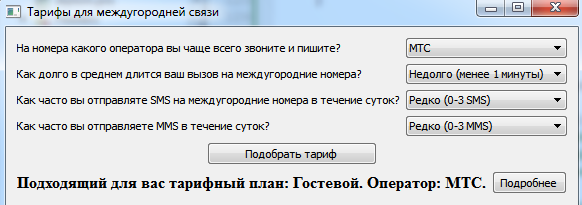


Рис 4.15. Вывод найденного тарифного плана

Нажав кнопку «Подробнее» пользователь может ознакомиться условиями подобранного по заданным параметрам тарифного плана (рис.4.16).

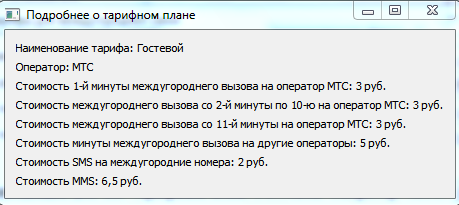


Рис 4.16. Условия найденного тарифного плана

## Выводы по главе 4

1. Разработана архитектура экспертной системы.
2. Разработана модель базы знаний, в которой будут храниться данные по тарифным планам, координаты 3G вышек и нечеткие правила для работы системы.
3. Разработан интерфейс пользователя, позволяющий организовать диалог, как с пользователем, так и с экспертом.
4. Проведено тестирование экспертной системы.

# ЗАКЛЮЧЕНИЕ

Основные результаты данной бакалаврской работы:

1. Выполнен анализ принципов построения экспертных систем.
2. Исследованы модели знаний, используемые в экспертных системах.
3. Разработана модель для представления знаний в разрабатываемой экспертной системе.
4. Разработан набор правил для экспертной системы, позволяющих выявить тарифный план, соответствующий предпочтениям пользователя.
5. Разработан метод определения коэффициента покрытия на основе механизма нечёткой логики.
6. Выполнена реализация и тестирование экспертной системы.

В данной работе были рассмотрены основные понятия нечеткой логики, модели знаний и экспертной системы. Была выбрана продукционная модель построения базы знаний, потому что она являются наиболее наглядным средством представления знаний. Для создания алгоритма определения подходящего тарифного плана был выбран механизм нечёткой логики, так как он обеспечивает представление о тарифах сотовых компаний в виде лингвистических переменных, значения которых может задавать пользователь на ограниченном естественном языке.

Был выполнен анализ предметной области тарифных планов сотовых компаний и предметной области покрытия 3G сигналом, после чего был разработан набор лингвистических переменных, характеризующих исходную информацию о тарифных планах, и была разработана лингвистическая переменная, характеризующая уровень приема 3G сигнала. Был разработан метод получения набора правил для разрабатываемой экспертной системы и метод вычисления коэффициента покрытия области 3G сигналом.

Также были разработаны: архитектура экспертной системы, модель ее базы знаний и интерфейс пользователя. Было проведено тестирование разработанной экспертной системы.

# БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Бланшет Ж., Семмерфилд М. Qt4 Программирование GUI на C++. – Пер. с англ. 2-е издание, доп. – М.: КУДИЦ-ПРЕСС, 2008 – 736 c.
2. Шлее М. Qt 4.8. Профессиональное программирование на С++.– СПб.: БХВ-Петербург, 2010 г. – 896 с.: ил. – (В подлиннике)
3. Аверкин А.Н., Батыршин И.З. Нечеткие множества в моделях управления и искусственного интеллекта. – М.: Наука, 1986 – 312с.
4. Вагин В.Н.,. Еремеев А.П. Некоторые базовые принципы построения интеллектуальных систем поддержки принятия решений реального времени // Изв. РАН. ТиСУ, 2001, № 6,с. 114-123.
5. Башмаков А.И. Интеллектуальные информационные технологии. - М.: Изд-во МГТУ им. Н. Э. Баумана, 2005. - 304 с: ил. — (Информатика в техническом университете).
6. Т.Тэрано, К.Асаи, М.Сугэно. Прикладные нечеткие системы. – М: Мир, 1993 – 368с.
7. Диго С. М. Базы данных. Проектирование и создание: Учебно-методический комплекс. – М.: Изд. центр ЕАОИ. 2008. – 171 с.
8. Администрирование Microsoft SQL Server 2000. Учебный курс MCSA/MCSE, MCDBA/Пер. с англ. – 2-е изд., испр. – М.: Издательско-торговый дом «Русская Редакция», 2002. — 640 стр.: ил.
9. Кнут Д. Искусство программирования, том 1. Основные алгоритмы, 3-е изд.: Пер. с англ. : Уч. пос. – М. : Издательский дом «Вильямс», 2011 – 720 с. : ил. – Парал. тит. англ.
10. Александреску А. Современное проектирование на С++. – М.: Вильямс, 2008, 968 с.
11. Муромцев Д.И. Введение в технологию экспертных систем. СПб: СПб ГУ ИТМО, 2005.
12. Попов Э.В. Экспертные системы: Решение неформализованных задач в диалоге с ЭВМ. - М.: Наука. Гл. ред. физ.-мат. Лит., 1987 г.
13. Андрейчиков А.В., Андрейчикова О.Н. Интеллектуальные информационные системы: М. Наука, 2004 г.
14. http://www.intuit.ru/ - официальный сайт НОУ «Интуит».
15. Яхъяева Г. Э. Нечеткие множества и нейронные сети. - М: Вильямс, 2006 – 315с.

# ПРИЛОЖЕНИЯ

## Приложение 1. Исходный код некоторых модулей программы

**// Вычисление коэффициента покрытия 3G сигнала для определенного оператора**

qreal forUser :: koef\_3G(QString oper){

file.setFileName("3G.txt");

file.*open*(QIODevice::WriteOnly);

QTextStream out(&file);

qreal gran\_sev\_sh\_1 = 54.32;

qreal gran\_sev\_sh\_2 = 55.29;

qreal gran\_vost\_dolg\_1 = 19.64;

qreal gran\_vost\_dolg\_2 = 22.89;

qreal kol\_point\_uver\_sign = 0;

qreal kol\_point\_sred\_sign = 0;

qreal all\_point = 0;

bool prenad\_oblast = false;

qreal i = gran\_sev\_sh\_1;

qreal j = gran\_vost\_dolg\_1;

while (i<=gran\_sev\_sh\_2){

while (j<=gran\_vost\_dolg\_2){

QPointF p(i,j);

prenad\_oblast = oblast.containsPoint(p,Qt::OddEvenFill);

if (prenad\_oblast){

all\_point++;

QVector<qreal> vec\_mu\_uver;

QVector<qreal> vec\_mu\_sred;

QVector<qreal> vec\_mu\_neuver;

if (oper =="Megafon"){

for (int k = 0; k < sotMegafon.size(); ++k){

QPointF sot = sotMegafon.at(k);

vec\_mu\_uver << 1/(1 + (i-sot.x())\*(i-sot.x()) + (j-sot.y())\*(j-sot.y()));

vec\_mu\_sred << 1/(1 + ((i-sot.x())-0.125)\*((i-sot.x())-0.125) + ((j-sot.y())-0.125)\*((j-sot.y())-0.125));

vec\_mu\_neuver << 1/(1 + ((i-sot.x())-0.5)\*((i-sot.x())-0.5) + ((j-sot.y())-0.5)\*((j-sot.y())-0.5));

}

}

if (oper =="MTS"){

for (int k = 0; k < sotMTS.size(); ++k){

QPointF sot = sotMTS.at(k);

vec\_mu\_uver << 1/(1 + (i-sot.x())\*(i-sot.x()) + (j-sot.y())\*(j-sot.y()));

vec\_mu\_sred << 1/(1 + ((i-sot.x())-0.125)\*((i-sot.x())-0.125) + ((j-sot.y())-0.125)\*((j-sot.y())-0.125));

vec\_mu\_neuver << 1/(1 + ((i-sot.x())-0.5)\*((i-sot.x())-0.5) + ((j-sot.y())-0.5)\*((j-sot.y())-0.5));

}

}

if (oper =="Beeline"){

for (int k = 0; k < sotBeeline.size(); ++k){

QPointF sot = sotBeeline.at(k);

vec\_mu\_uver << 1/(1 + (i-sot.x())\*(i-sot.x()) + (j-sot.y())\*(j-sot.y()));

vec\_mu\_sred << 1/(1 + ((i-sot.x())-0.125)\*((i-sot.x())-0.125) + ((j-sot.y())-0.125)\*((j-sot.y())-0.125));

vec\_mu\_neuver << 1/(1 + ((i-sot.x())-0.5)\*((i-sot.x())-0.5) + ((j-sot.y())-0.5)\*((j-sot.y())-0.5));

}

}

if ((\*std::max\_element(vec\_mu\_uver.begin(),vec\_mu\_uver.end()) >= \*std::max\_element(vec\_mu\_sred.begin(),vec\_mu\_sred.end())) || (\*std::max\_element(vec\_mu\_uver.begin(),vec\_mu\_uver.end()) > \*std::max\_element(vec\_mu\_neuver.begin(),vec\_mu\_neuver.end()))) kol\_point\_uver\_sign++;

if ((\*std::max\_element(vec\_mu\_sred.begin(),vec\_mu\_sred.end()) >= \*std::max\_element(vec\_mu\_neuver.begin(),vec\_mu\_neuver.end())) || (\*std::max\_element(vec\_mu\_sred.begin(),vec\_mu\_sred.end()) > \*std::max\_element(vec\_mu\_uver.begin(),vec\_mu\_uver.end()))) kol\_point\_sred\_sign++;

}

j+=0.01;

}

j = gran\_vost\_dolg\_1;

i+=0.01;

}

out << all\_point << "\r\n";

out << (kol\_point\_uver\_sign + kol\_point\_sred\_sign/3)/all\_point << "\r\n" ;

return (kol\_point\_uver\_sign + kol\_point\_sred\_sign/3)/all\_point;

}

**// Перевод условий местных тарифных планов в нечеткие условия**

void table\_tarif :: btn\_renew(){

QSqlQuery query1;

QSqlQuery query;

QSqlQuery query2;

query.exec("TRUNCATE TABLE mestn\_tarif");

query1.exec("SELECT \* FROM Тарифный\_план\_для\_\_местной\_связи");

while(query1.next()){

QString tarif = query1.value(0).toString();

int oper = query1.value(1).toInt();

qreal min1 = query1.value(2).toDouble();

qreal min2\_10 = query1.value(4).toDouble();

qreal min11 = query1.value(5).toDouble();

qreal min\_dif = query1.value(6).toDouble();

qreal sms = query1.value(3).toDouble();

qreal mms = query1.value(7).toDouble();

QString oper\_str;

if (oper == 1) oper\_str = "МТС";

if (oper == 2) oper\_str = "Билайн";

if (oper == 3) oper\_str = "Мегафон";

qreal mu\_vys\_min1 = 1/(1+(2-min1)\*(2-min1));

qreal mu\_sred\_min1 = 1/(1+(1-min1)\*(1-min1));

qreal mu\_nizk\_min1 = 1/(1+(0-min1)\*(0-min1));

qreal mu\_vys\_min2\_10 = 1/(1+(2-min2\_10)\*(2-min2\_10));

qreal mu\_sred\_min2\_10 = 1/(1+(1-min2\_10)\*(1-min2\_10));

qreal mu\_nizk\_min2\_10 = 1/(1+(0-min2\_10)\*(0-min2\_10));

qreal mu\_vys\_min11 = 1/(1+(2-min11)\*(2-min11));

qreal mu\_sred\_min11 = 1/(1+(1-min11)\*(1-min11));

qreal mu\_nizk\_min11 = 1/(1+(0-min11)\*(0-min11));

qreal mu\_vys\_min\_dif = 1/(1+(2-min\_dif)\*(2-min\_dif));

qreal mu\_sred\_min\_dif = 1/(1+(1-min\_dif)\*(1-min\_dif));

qreal mu\_nizk\_min\_dif = 1/(1+(0-min\_dif)\*(0-min\_dif));

qreal mu\_vys\_sms = 1/(1+(2-sms)\*(2-sms));

qreal mu\_sred\_sms = 1/(1+(1-sms)\*(1-sms));

qreal mu\_nizk\_sms = 1/(1+(0-sms)\*(0-sms));

qreal mu\_vys\_mms = 1/(1+(10-mms)\*(10-mms));

qreal mu\_sred\_mms = 1/(1+(8-mms)\*(8-mms));

qreal mu\_nizk\_mms = 1/(1+(6-mms)\*(6-mms));

QString znach\_min1;

QString znach\_min2\_10;

QString znach\_min11;

QString znach\_min\_dif;

QString znach\_sms;

QString znach\_mms;

if(qMax(qMax(mu\_vys\_min1,mu\_sred\_min1),mu\_nizk\_min1) == mu\_vys\_min1) znach\_min1 = "высокая";

if(qMax(qMax(mu\_vys\_min1,mu\_sred\_min1),mu\_nizk\_min1) == mu\_sred\_min1) znach\_min1 = "средняя";

if(qMax(qMax(mu\_vys\_min1,mu\_sred\_min1),mu\_nizk\_min1) == mu\_nizk\_min1) znach\_min1 = "низкая";

if(qMax(qMax(mu\_vys\_min2\_10,mu\_sred\_min2\_10),mu\_nizk\_min2\_10) == mu\_vys\_min2\_10) znach\_min2\_10 = "высокая";

if(qMax(qMax(mu\_vys\_min2\_10,mu\_sred\_min2\_10),mu\_nizk\_min2\_10) == mu\_sred\_min2\_10) znach\_min2\_10 = "средняя";

if(qMax(qMax(mu\_vys\_min2\_10,mu\_sred\_min2\_10),mu\_nizk\_min2\_10) == mu\_nizk\_min2\_10) znach\_min2\_10 = "низкая";

if(qMax(qMax(mu\_vys\_min11,mu\_sred\_min11),mu\_nizk\_min11) == mu\_vys\_min11) znach\_min11 = "высокая";

if(qMax(qMax(mu\_vys\_min11,mu\_sred\_min11),mu\_nizk\_min11) == mu\_sred\_min11) znach\_min11 = "средняя";

if(qMax(qMax(mu\_vys\_min11,mu\_sred\_min11),mu\_nizk\_min11) == mu\_nizk\_min11) znach\_min11 = "низкая";

if(qMax(qMax(mu\_vys\_min\_dif,mu\_sred\_min\_dif),mu\_nizk\_min\_dif) == mu\_vys\_min\_dif) znach\_min\_dif = "высокая";

if(qMax(qMax(mu\_vys\_min\_dif,mu\_sred\_min\_dif),mu\_nizk\_min\_dif) == mu\_sred\_min\_dif) znach\_min\_dif = "средняя";

if(qMax(qMax(mu\_vys\_min\_dif,mu\_sred\_min\_dif),mu\_nizk\_min\_dif) == mu\_nizk\_min\_dif) znach\_min\_dif = "низкая";

if(qMax(qMax(mu\_vys\_sms,mu\_sred\_sms),mu\_nizk\_sms) == mu\_vys\_sms) znach\_sms = "высокая";

if(qMax(qMax(mu\_vys\_sms,mu\_sred\_sms),mu\_nizk\_sms) == mu\_sred\_sms) znach\_sms = "средняя";

if(qMax(qMax(mu\_vys\_sms,mu\_sred\_sms),mu\_nizk\_sms) == mu\_nizk\_sms) znach\_sms = "низкая";

if(qMax(qMax(mu\_vys\_mms,mu\_sred\_mms),mu\_nizk\_mms) == mu\_vys\_mms) znach\_mms = "высокая";

if(qMax(qMax(mu\_vys\_mms,mu\_sred\_mms),mu\_nizk\_mms) == mu\_sred\_mms) znach\_mms = "средняя";

if(qMax(qMax(mu\_vys\_mms,mu\_sred\_mms),mu\_nizk\_mms) == mu\_nizk\_mms) znach\_mms = "низкая";

query2.exec("INSERT INTO mestn\_tarif VALUES('" + oper\_str + "','" + tarif +

"','" + znach\_min1 + "','" + znach\_min2\_10 + "','" + znach\_min11 +

"','" + znach\_min\_dif + "','" + znach\_sms + "','" + znach\_mms + "')");

}

}

**// Поиск местного тарифного плана, соответствующего заданным пользователем параметрам**

void local\_connection :: baseRulesForLocal(int ansOperator, int ansLong, int ansSMS, int ansMMS){

//МТС

if(ansOperator==1&&ansLong==1){

QSqlQuery query;

query.exec("SELECT name FROM mestn\_tarif WHERE operator ='МТС' AND min\_1='низкая'");

if(query.next())lb5->setText("Оператор: МТС; Тариф: "+query.value(0).toString());

else lb5->setText("Соответствий не найдено");

}

if(ansOperator==1&&ansLong==2){

QSqlQuery query;

query.exec("SELECT name FROM mestn\_tarif WHERE operator ='МТС' AND min2\_10='низкая'");

if(query.next())lb5->setText("Оператор: МТС; Тариф: "+query.value(0).toString());

else lb5->setText("Соответствий не найдено");

}

if(ansOperator==1&&ansLong==3){

QSqlQuery query;

query.exec("SELECT name FROM mestn\_tarif WHERE operator ='МТС' AND min\_11='низкая'");

if(query.next())lb5->setText("Оператор: МТС; Тариф: "+query.value(0).toString());

else lb5->setText("Соответствий не найдено");

}

//Билайн

if(ansOperator==2&&ansLong==1){

QSqlQuery query;

query.exec("SELECT name FROM mestn\_tarif WHERE operator ='Билайн' AND min\_1='низкая'");

if(query.next())lb5->setText("Оператор: Билайн; Тариф: "+query.value(0).toString());

else lb5->setText("Соответствий не найдено");

}

if(ansOperator==2&&ansLong==2){

QSqlQuery query;

query.exec("SELECT name FROM mestn\_tarif WHERE operator ='Билайн' AND min2\_10='низкая'");

if(query.next())lb5->setText("Оператор: Билайн; Тариф: "+query.value(0).toString());

else lb5->setText("Соответствий не найдено");

}

if(ansOperator==2&&ansLong==3){

QSqlQuery query;

query.exec("SELECT name FROM mestn\_tarif WHERE operator ='Билайн' AND min\_11='низкая'");

if(query.next())lb5->setText("Оператор: Билайн; Тариф: "+query.value(0).toString());

else lb5->setText("Соответствий не найдено");

}

//Мегафон

if(ansOperator==3&&ansLong==1){

QSqlQuery query;

query.exec("SELECT name FROM mestn\_tarif WHERE operator ='Мегафон' AND min\_1='низкая'");

if(query.next())lb5->setText("Оператор: Мегафон; Тариф: "+query.value(0).toString());

else lb5->setText("Соответствий не найдено");

}

if(ansOperator==3&&ansLong==2){

QSqlQuery query;

query.exec("SELECT name FROM mestn\_tarif WHERE operator ='Мегафон' AND min2\_10='низкая'");

if(query.next())lb5->setText("Оператор: Мегафон; Тариф: "+query.value(0).toString());

else lb5->setText("Соответствий не найдено");

}

if(ansOperator==3&&ansLong==3){

QSqlQuery query;

query.exec("SELECT name FROM mestn\_tarif WHERE operator ='Мегафон' AND min\_11='низкая'");

if(query.next())lb5->setText("Оператор: Мегафон; Тариф: "+query.value(0).toString());

else lb5->setText("Соответствий не найдено");

}

//Все операторы

if(ansOperator==4){

QSqlQuery query;

query.exec("SELECT operator,name FROM mestn\_tarif WHERE min\_dif = 'низкая'");

if(query.next())lb5->setText("Оператор: "+query.value(0).toString()+"; Тариф: "+query.value(1).toString());

else lb5->setText("Соответствий не найдено");

}

}

void local\_connection :: btnFindTariForLocal(){

if(cmb1->currentText()=="МТС") answerOperator = 1;

if(cmb1->currentText()=="Билайн") answerOperator = 2;

if(cmb1->currentText()=="Мегафон") answerOperator = 3;

if(cmb1->currentText()=="Одинаково на все") answerOperator = 4;

if (cmb2->currentText()=="Недолго (менее 1 минуты)") answerLong = 1;

if (cmb2->currentText()=="Умеренно (2-10 минут)") answerLong = 2;

if (cmb2->currentText()=="Долго (более 10 мин)") answerLong = 3;

if(cmb3->currentText()== "Редко (0-3 SMS)")answerSMS = 1;

if(cmb3->currentText()== "Умеренно (4-10 SMS)")answerSMS = 2;

if(cmb3->currentText()== "Часто (более 10 SMS)")answerSMS = 3;

if(cmb4->currentText()== "Редко (0-3 MMS)")answerMMS = 1;

if(cmb4->currentText()== "Умеренно (4-10 MMS)")answerMMS = 2;

if(cmb4->currentText()== "Часто (более 10 MMS)")answerMMS = 3;

baseRulesForLocal(answerOperator, answerLong, answerSMS, answerMMS);

}