

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Тольяттинский государственный университет»

Институт математики, физики и информационных технологий

(наименование института полностью)

Кафедра «Прикладная математика и информатика»

(наименование кафедры)

01.03.02 Прикладная математика и информатика

(код и наименование направления подготовки, специальности)

Системное программирование и компьютерные технологии

(направленность (профиль)/специализация)

БАКАЛАВРСКАЯ РАБОТА

на тему Методы критериального анализа в решении задачи выбора поставщика

Студент

А.В. Курушина

(И.О. Фамилия)

(личная подпись)

Руководитель

Э.В. Егорова

(И.О. Фамилия)

(личная подпись)

Допустить к защите

Заведующий кафедрой к.т.н., доцент, А.В. Очеповский

(ученая степень, звание, И.О. Фамилия)

(личная подпись)

« _____ » _____ 20 _____ г.

Тольятти 2018

АННОТАЦИЯ

Актуальность темы данной работы обусловлена проблемой выбора среди различных альтернатив действий и выборе оптимальной альтернативы, т. е. такой, которая позволяет получить наилучший результат в достижении поставленной цели. Такое явление встречается ежедневно и повсеместно. В рамках данной выпускной квалификационной работы рассмотрена проблема выбора поставщика для ресторанной компании.

Выпускная квалификационная работа состоит из 47 страниц. В работе: рисунков – 6, таблиц – 23, приложений – 1, список литературы состоит из 20 источников.

Выпускная квалификационная работа посвящена разработке решения задачи о выборе поставщика с использованием методов критериального анализа. Она состоит из введения, трех глав, заключения и приложения.

Во введении обосновывается актуальность темы исследования, формулируются цели и задачи, которые необходимо решить для достижения поставленной цели.

В первой главе поставлена задача, представляющая собой выбор поставщика. Затем проанализированы существующие методы критериального анализа и выбран тот, который является наиболее подходящим для решения задачи выбора поставщика.

Во второй главе представлено решение задачи выбора поставщика посредством математического анализа. Далее представлены требования к программе, которая должна реализовать математическую модель выбранного метода, и непосредственно работа самой программы.

В третьей главе приведено решение задачи выбранным методом, но в более сложных условиях. И доказана оптимальность выбранного решения.

В заключении сделаны основные выводы и итоги по проделанной работе.

ABSTRACT

The relevance of the topic of this work is due to the problem of choosing among the various alternatives of actions and the choice of the optimal alternative, i.e., one that gives the best result in achieving the goal. This phenomenon occurs daily and everywhere. Within the framework of this final qualifying work, the problem of choosing a supplier for a restaurant company was considered.

Graduation work consists of 47 pages in the work: figures – 6, tables – 23, applications – 1, the list of literature consists of 20 sources.

The is devoted to the development of the solution to the problem of choosing a supplier using the methods of criterial analysis. It consists of an introduction, three chapters, a conclusion and an appendix.

In the introduction, the urgency of the research topic is substantiated, goals and tasks are formulated, which must be solved to achieve the goal.

The first chapter sets the task, which is the choice of the supplier. Then, the existing methods of criterial analysis are analyzed and the one that is most suitable for solving the problem of selecting a supplier is selected.

The second chapter presents a solution to the problem of choosing a supplier through mathematical analysis. Next, the requirements for the program are presented, which should implement the mathematical model of the chosen method and the work of the program itself.

The third chapter presents a solution to the problem by the method chosen earlier, but in more difficult conditions that are often encountered in life than those mentioned in Chapter 2.

In conclusion, the main conclusions and conclusions on the work done.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	6
1 ПОСТАНОВКА ЗАДАЧИ ВЫБОРА СРЕДИ НЕСКОЛЬКИХ АЛЬТЕРНАТИВ И АНАЛИЗ СУЩЕСТВУЮЩИХ МЕТОДОВ КРИТЕРИАЛЬНОГО АНАЛИЗА	8
1.1 Постановка задачи выбора среди нескольких альтернатив	8
1.2 Анализ методов решения задачи выбора в условиях полной определенности	11
1.2.1 Анализ метода равномерной оптимизации	11
1.2.2 Анализ метода справедливого компромисса	11
1.2.3 Анализ метода свертывания критериев	12
1.2.4 Анализ метода главного критерия	13
1.2.5 Анализ метода идеальной точки	13
1.3 Анализ методов решения задачи выбора в условиях неопределенности и риска	14
1.3.1 Анализ аксиоматического метода	15
1.3.2 Анализ прямых методов	16
1.3.3 Анализ метода компенсации	17
1.3.4 Анализ метода анализа иерархий	18
1.4 Анализ методов критериального анализа	19
1.5 Алгоритмическое решение задачи выбора поставщика для ресторанной компании	21
2 РЕШЕНИЕ ЗАДАЧИ КРИТЕРИАЛЬНОГО АНАЛИЗА О ВЫБОРЕ ПОСТАВЩИКА ДЛЯ РЕСТОРАННОЙ КОМПАНИИ	23
2.1 Решение задачи о выборе среди нескольких альтернатив методом анализа иерархий	23
2.2 Решение задачи о выборе среди нескольких альтернатив посредством математического анализа	27

2.3	Создание программы, реализующей математическую модель метода анализа иерархий.....	35
2.4	Проверка правильности работы программы, реализующей метод анализа иерархий.....	38
3	ОПРЕДЕЛЕНИЕ ОПТИМАЛЬНОГО РЕШЕНИЯ В ВЫБОРЕ ПОСТАВЩИКА С ИСПОЛЬЗОВАНИЕМ МЕТОДА АНАЛИЗА ИЕРАРХИЙ.....	40
3.1	Исходные данные для задачи о выборе поставщика	40
3.2	Решение задачи выбора поставщика с использованием метода анализа иерархий.....	40
	ЗАКЛЮЧЕНИЕ	44
	СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ.....	45
	ПРИЛОЖЕНИЕ А	48

ВВЕДЕНИЕ

Проблема выбора заключается в определении различных альтернатив действий и выборе оптимальной альтернативы, т. е. такой, которая позволяет получить наилучший результат в достижении поставленной цели.

В качестве альтернатив могут выступать новые целевые области (товарные рынки), виды выпускаемой продукции, инвестиции в различные сферы деятельности предприятия и т. д. Как правило, они не могут быть реализованы одновременно.

Целенаправленный выбор среди подобных альтернатив представляет собой принятие такого решения, которое предоставило бы как можно больше преимуществ [21].

Реализация (осуществление) любой возможной альтернативы ведет к одному или нескольким последствиям (результатам). Ожидаемыми результатами могут быть выручка от реализации товаров, издержки производства, доля удовлетворения спроса, прибыль, затраты на продвижение товара, доля рынка и др [20].

Таким образом, актуальность темы выпускной квалификационной работы (ВКР) обусловлена необходимостью разработки программы, которая предоставляет определенные альтернативы и показывает, какая из них является оптимальной для лица, принимающего решения.

Объектом исследования является технология выбора поставщика для ресторанной компании.

Предметом исследования являются методы и алгоритмы критериального анализа при решении задачи о выборе.

Целью выпускной квалификационной работы является разработка решения задачи о выборе поставщика с использованием методов критериального анализа.

Задачи выпускной квалификационной работы:

- рассмотреть методы критериального анализа;

- выбрать метод критериального анализа для решения задачи о выборе поставщика;
- реализовать выбранный метод критериального анализа путем математического анализа;
- разработать программу, реализующую математическую модель выбранного метода;
- определить оптимальное решение в выборе поставщика с использованием выбранного метода.

В первой главе поставлена задача, представляющая собой выбор поставщика. Затем проанализированы существующие методы критериального анализа и выбран тот, который является наиболее подходящим для решения задачи выбора поставщика.

Во второй главе представлено решение задачи выбора поставщика посредством математического анализа. Далее представлены требования к программе, которая должна реализовать математическую модель выбранного метода и непосредственно работа самой программы.

В третьей главе приведено решение задачи выбранным ранее методом, но в более сложных условиях, которые часто встречаются в жизни, нежели те, которые были упомянуты в главе 2.

1 ПОСТАНОВКА ЗАДАЧИ ВЫБОРА СРЕДИ НЕСКОЛЬКИХ АЛЬТЕРНАТИВ И АНАЛИЗ СУЩЕСТВУЮЩИХ МЕТОДОВ КРИТЕРИАЛЬНОГО АНАЛИЗА

Все подходы критериального делают варианты и их вклад в различные критерии явными, и все они реализуются, учитывая некоторые суждения и предпочтения. Однако они различаются в том, как они объединяют данные. Формальные методы критериального анализа обычно предоставляют явную систему относительного взвешивания для разных критериев.

Методы критериального анализа могут использоваться для определения одного наиболее предпочтительного варианта, для ранжирования опций, для краткого списка ограниченного числа вариантов последующей оценки или просто для того, чтобы отличить приемлемые от неприемлемых возможностей. Существует много методов критериального анализа и их число все еще растет.

Основная роль этих методов заключается в том, чтобы справиться с трудностями, которые возникают у лиц, принимающих решения в области прав человека, для обработки большого количества сложной информации и т.д. Данные методы делятся на те, которые работают с чёткой информацией, а также с нечёткой информацией в условиях неопределенности и риска. В этой главе представлен краткий анализ некоторых методов критериального анализа.

1.1 Постановка задачи выбора среди нескольких альтернатив

Задача выбора состоит в том, что необходимо принять такое решение, которое гарантировало бы предприятию не только прибыль, но и избавило бы от убытков. Таких задач может быть множество в зависимости от компании, например: в аэропорту необходимо составить оптимальное расписание, в благотворительном фонде распределить денежные средства и т.д. Данные задачи решает такая область математического анализа как критериальный анализ.

Задача критериального анализа состоит в выборе лучшей альтернативы среди нескольких предложенных по определенным критериям. Критериев

может быть несколько, также они могут иметь разные единицы измерения, а показатели непостоянны [7].

В самой общей форме любая задача выбора среди нескольких альтернатив может быть представлена в виде «дано...», «требуется определить...». Руководствуясь этой формой, формальная модель задачи выбора среди нескольких альтернатив для индивидуального лица, принимающего решение, (ЛПР) может быть описана следующим образом:

$$\langle S_0, T, Q \mid S, A, B, Y, f, K, Y_{opt} \rangle,$$

где слева от вертикальной черты расположены известные, а справа – неизвестные элементы задачи:

S_0 – проблемная ситуация;

T – время для принятия решения;

Q – имеющиеся для принятия решения ресурсы;

$S = (S_1, \dots, S_n)$ – множество альтернативных ситуаций, уточняющих проблемную ситуацию S_0 ;

$A = (A_1, \dots, A_k)$ – множество целей, преследуемых при принятии решения;

$B = (B_1, \dots, B_L)$ – множество ограничений;

$Y = (Y_1, \dots, Y_m)$ – множество альтернативных вариантов решения;

f – функция предпочтения ЛПР;

K – критерий выбора наилучшего решения;

Y_{opt} – оптимальное решение [2].

Таким образом, содержание задачи о выборе среди нескольких альтернатив можно сформулировать следующим образом: в условиях проблемной ситуации S_0 , располагаемого времени T и ресурсов Q необходимо уточнить ситуацию S_0 множеством гипотетических ситуаций S , сформировать множества целей A , ограничений B и альтернативных вариантов решения Y , произвести оценку индивидуальных предпочтений решений f и найти оптимальное решение Y_{opt} , руководствуясь сформулированным критерием выбора K – для индивидуального ЛПР [8].

В случае ресторанной компании, одной из задач выбора является выбор поставщика, который будет поставлять продукцию в заведение. Исходя из модели задачи выбора среди нескольких альтернатив, можно составить модель задачи выбора поставщика:

- S_0 – выбор поставщика;
- T – время для принятия решения;
- Q – денежные средства;
- $S = (S_1, \dots, S_n)$ – покупка товаров у одного поставщика или же у нескольких поставщиков одновременно;
- $A = (A_1, \dots, A_k)$ – за минимальное время сделать оптимальный выбор, гарантирующий прибыль предприятию;
- $B = B_1, \dots, B_L$ – денежные средства, которыми располагает предприятие, время, в течение которого должен быть осуществлен выбор поставщика и т.д.;
- $Y = (Y_1, \dots, Y_m)$ – покупаются товары у одного поставщика или сразу у нескольких, одновременно это делается или же сначала закупаются одни товары, а затем другие и т.д.;
- f – что является более важным для ЛПР: цена, качество или время на осуществление выбора и т.д.;
- K – определяется на основе функции предпочтений f ;
- Y_{opt} – непосредственно выбор поставщика, который бы подошел ЛПР, учитывая все его предпочтения.

В случае задачи выбора поставщика возможны два варианта событий: либо имеется полная определенность: сколько конкретно предполагается потратить денег на товар, какое время готов ждать заказчик и т.д., либо все показатели постоянно варьируются. Таким образом имеется два варианта развития событий:

- выбор в условиях полной определенности;
- выбор в условиях неопределенности и риска.

Далее будут рассмотрены методы решения задачи выбора в условиях полной определенности, а также в условиях неопределенности и риска.

1.2 Анализ методов решения задачи выбора в условиях полной определенности

1.2.1 Анализ метода равномерной оптимизации

Исходной посылкой (принципом) данного метода является то, что все критерии оптимальности считаются экономически равноценными. Согласно этому методу лучшим считается вариант, у которого суммарная величина всех числовых значений целевых функций принимает максимальное значение. Т.е. основная формула для данного метода имеет вид:

$$f(A^*) = \max_i \sum_{j=1}^n f_j A_i \quad (1)$$

где i – код целевой функции ($i = 1, n$);

j – номер альтернативного варианта ($j = 1, m$);

A_i – множество j -х вариантов.

Метод равномерной оптимизации применяется, если глобальное качество альтернативы представляет собой сумму локальных (частных) качеств и, кроме того, все критерии имеют одну и ту же единицу измерения, например, денежное выражение либо безразмерные величины [8].

Главный недостаток метода – это возможность компенсации малых значений некоторых критериев достаточно большими значениями других. Также рассматриваемый метод неактуален в случае, если задача имеет несколько критериев с разными единицами измерения.

1.2.2 Анализ метода справедливого компромисса

Выбор лучшей альтернативы определяется на основе максимального значения произведения анализируемых критериев.

$$f(A^*) = \max_i \sum_{j=1}^n f_j A_i \quad (2)$$

где A_i – множество допустимых значений вектора варьируемых параметров i .

Данный метод используется в случае, когда все критерии одинаково важны. Однако, далеко не в каждой задаче все критерии одинаково важны, следовательно, можно сделать вывод, что данный метод неактуален, т.к. ориентирован на малый спектр задач.

1.2.3 Анализ метода свертывания критериев

Решение задач выбора на основе линейной свертки критериев состоит в назначении тем или иным способом неотрицательных (а чаще положительных) коэффициентов a_1, a_2, \dots, a_n , в сумме дающих единицу (хотя это не обязательно), и последующей максимизации линейной комбинации критериев.

$$f^*(a_i) = \sum_{j=1}^n a_j f_j^* \rightarrow \max, \quad \sum_{j=1}^n a_j = 1 \quad (3)$$

где a_i – коэффициент важности по i -му критерию;

$f_j^* a_i$ – численное значение нормализованного i -го критерия j -го варианта.

В этом случае каждому из анализируемых критериев присваивается весовой коэффициент, который отражает его степень значимости в системе целей предприятия. Конечным результатом является один наиболее важный по результатам анализа критерий.

Но в случае с несколькими показателями возникает проблема свертывания критериев. Однако нельзя сказать, что данный метод не подходит для решения поставленной задачи, но если один из последующих методов окажется оптимальнее по отношению к методу свертывания критериев, то рассматриваемый в данный момент метод будет отброшен.

1.2.4 Анализ метода главного критерия

В этом случае устанавливается наиболее значимый критерий (на основе экспертных оценок), для которого выбирается максимальное значение. Остальные критерии рассматриваются как ограничения в решении задачи.

$$f_j^* = f_j^* \text{ а } \rightarrow \max, f_j^* \geq D_j \quad (4)$$

где D_j – нижняя граница, соответствующая какому-либо предельному значению.

Иными словами, на основе выполненных выше оценок (принципов оптимальности) разрабатывается экономико-математическая модель, в которой могут получить отражение различные варианты нормализации показателей, а также альтернативы выбора принципа оптимальности.

Данный метод плох тем, что далеко не в каждой задаче присутствуют экспертные оценки, а тем более в задачах с нечеткой информацией. Таким образом, не всегда легко выбрать главный критерий, а иногда и вовсе невозможно.

1.2.5 Анализ метода идеальной точки

Ищется план, удовлетворяющий условию

$$A_i = \sum_{i=1}^n W_i I_i - X_i \quad (5)$$

где A – отношение избирателя к претенденту;

W – показатель важности i -й характеристики претендента;

I – «идеальное» значение i -й характеристики претендента;

X – мнение о фактической величине i -й характеристики претендента;

n – число i -х характеристик претендента.

На первом этапе применяется метод нормализации критериев (получение безразмерной формы критериев). Так как критерии могут иметь различные масштабы и шкалы измерения, то прежде, чем приступить к решению задачи выбора, их необходимо привести к одной единице измерения (обычно к безразмерному виду). Этот процесс называется нормализацией. Существуют

различные методы нормализации, Предлагается следующий способ получения безразмерной формы критериев:

$$f_j^n A_i = \frac{f_j A_i - \min_i f_j A_i}{\max_i f_j A_i - \min_j f_j A_i} \quad (6)$$

где $j = 1, n, \min_j \{f_j A_i\} \neq \max_i \{f_i A_j\}$.

Данный метод имеет ряд недостатков таких как неоднозначность выбора метрики; невыполнимость аксиомы независимости. Кроме того, может случиться такое, что худшая альтернатива станет доминировать над лучшей альтернативой.

Все рассмотренные выше методы предполагают работу в условиях полной определенности, что является довольно редким явлением в реальной жизни. Таким образом, имеет смысл рассмотреть методы предполагающие решение задачи о выборе в условиях неопределенности и риска.

1.3 Анализ методов решения задачи выбора в условиях неопределенности и риска.

Неопределенность – это состояние ограниченного знания текущих условий или будущих результатов. Это важный компонент риска, который связан с вероятностью и масштабом негативных последствий. Менеджеры часто сталкиваются с неопределенностью в своей работе. Чтобы минимизировать риск того, что их решения приведут к нежелательным результатам, они должны развивать навыки и суждения, необходимые для снижения этой неопределенности. Управление неопределенностью и риском также включает смягчение или даже удаление вещей, которые препятствуют эффективному принятию решений или неблагоприятно влияют на производительность.

Причинами неопределенности могут являться поведение конкурентов, персонала организации, технические и технологические процессы и изменения конъюнктурного характера, близость предстоящего события: вещи, которые

должны вскоре произойти, легче оценить, чем те, которые произойдут в дальнейшем будущем. Один из подходов к решению проблемы неопределенности заключается в том, чтобы отложить принятие решений, пока данные не станут более доступными и надежными. Конечно, отсрочка некоторых решений может привести к собственному набору рисков, особенно когда потенциальные негативные последствия ожидания велики [3].

Методы для работы в условиях неопределенности и риска несколько отличаются от методов, рассмотренных выше. Учитывается то, что критерии могут сильно варьироваться, а некоторые критерии могут и вовсе быть неизвестны. Ниже будут рассмотрены и проанализированы несколько методов критериального анализа нечеткой информации.

1.3.1 Анализ аксиоматического метода

Аксиоматический метод, в логике, процедура, посредством которой вся целая система (например, наука) генерируется в соответствии с указанными правилами путем логического вывода из некоторых основных положений (аксиом или постулатов), которые, в свою очередь, строятся из нескольких терминов, принятых в качестве начальных данных. Эти термины и аксиомы могут быть либо произвольно определены и построены, либо задуманы в соответствии с моделью, в которой, как считается, существует какой-то интуитивный критерий, указывающий на их истинность. Самыми старыми примерами аксиоматизированных систем являются силлогистика Аристотеля и геометрия Евклида.

Все методы оценки критериальных альтернатив, так или иначе, используют измерение полезности, но аксиоматические методы подходят к этим измерениям наиболее теоретически: они рассматривают их как определенные шаги, подтверждающие справедливость выбора некоторых аксиом и ведущие к возможности использования определенной функции полезности.

Аксиоматические методы непосредственно опираются на теорию ожидаемой полезности фон Неймана и Morgenштерна либо на более современные критериальной теории полезности.

Следует отметить искусственность аксиоматических методов. Главная практическая проблема, – многокритериальное сравнение альтернатив, – отступает на второй план перед чисто формальной (математической) проблемой поиска функции полезности в определенной форме [1].

1.3.2 Анализ прямых методов

Здесь зависимость общей полезности альтернативы от оценок по отдельным критериям считается известной заранее.

Существует большое число методов, в которых форма зависимости результирующей полезности альтернативы от ее оценок по многим критериям задается без всяких теоретических оснований, а параметры этой зависимости задаются так:

- нормативно, на основании утверждённых методик («закон есть закон»);
- непосредственно устанавливаются лицом принимающим решение («начальник всегда прав») или этим лицом и консультантами («система коллективной ответственности»);
- определяются из статистики, если она есть в достаточном объёме и надёжна («статистика знает всё»).

К основным методам данной группы относятся следующие: метод взвешенных сумм и его аналоги, метод матрицы решений, метод дерева решений и ряд других [1].

Данный метод хорош тем, что лицо, принимающее решение, может установить удобные для него критерии и параметры. В конце будет получен результат с учетом предпочтений, которые были указаны на начальной стадии.

Таким образом, данный метод можно использовать для разных заказчиков с разными предпочтениями.

1.3.3 Анализ метода компенсации

Рациональная модель принятия решений, в которой выбор систематически оценивается по различным критериям. Более выгодные атрибуты альтернативы могут компенсировать менее выгодные. Для этого необходимо следовать систематической процедуре принятия решений компенсирующей модели, поскольку положительный балл по одному атрибуту может перевесить отрицательный результат по другому атрибуту.

Решение компенсации подразумевает, что некоторый продавец «торгует» хорошими и плохими атрибутами продукта. Например, автомобиль может иметь низкую цену и малый пробег, но медленное ускорение. Либо автомобиль имеет быстрое ускорение, но в таком случае цена будет выше, и топлива он будет потреблять больше.

Иногда решение будет включать в себя некомпенсирующую стратегию. Например, производитель может отказаться от всех безалкогольных напитков, которые содержат искусственные подсластители. Здесь другие хорошие функции, такие как вкус и низкие калории, этот атрибут «не подлежит обсуждению».

Объем усилий, которые потребитель ставит на поиск, зависит от ряда таких факторов, как рынок (количество конкурентов и насколько велики различия между брендами, которые, как ожидается, будут), характеристики продукта (важность продукта, его сложность, очевидность его качества), потребительские характеристики (интерес потребителя, правильность анализа характеристик продукта и совершения наилучшей возможной сделки) и т.д.

По идее, это наиболее простой метод, при котором человек выписывает достоинства и недостатки каждой из альтернатив и, вычеркивая попарно эквивалентные достоинства (недостатки), изучает то, что осталось и делает выбор.

Переход от сравнения качеств по различным критериям к сравнению альтернатив может быть осуществлен различными путями. Среди них следует выделить построение кривых безразличия и сравнение разностей оценок альтернатив по критериям.

Методы построения кривых или поверхностей безразличия очень трудоемки и малопригодны при количестве критериев $n \geq 3$. Для оставшихся альтернатив проще использовать метод анализа иерархий (МАИ) или один из прямых методов [1].

1.3.4 Анализ метода анализа иерархий

Метода анализа иерархий (МАИ) был разработан для оптимизации принятия решений, когда перед человеком стоит сочетание качественных, количественных и иногда противоречивых факторов, которые учитываются в ходе анализа. МАИ очень эффективен в принятии сложных, зачастую необратимых решений.

МАИ использует решения лиц, принимающих решения, для разложения проблем в иерархии. Сложность представляется количеством уровней в иерархии, построенной с учётом модели принятия решений. Иерархия используется для получения отношений между объектами, а также относительной ценности. МАИ использует матричную алгебру для определения коэффициентов, чтобы получить математически оптимальное решение. МАИ использовался в многомиллиардных решениях.

МАИ выводит коэффициенты соотношения парных сравнений факторов и вариантов выбора. Типичные ситуации, в которых используется МАИ, относятся к:

- установке приоритетов факторов и требований, которые влияют на разработку и производительность программного обеспечения;
- выбору среди нескольких стратегий улучшения функций безопасности в автомобилях;

- выбору необходимых программных компонентов у нескольких поставщиков программного обеспечения;
- оценке качества исследовательских или инвестиционных предложений и т.д. [17].

Метод анализа иерархий на сегодняшний день является наиболее оптимальным, распространенным и удобным. С его помощью, не затрачивая больших усилий, можно принимать решения в условиях риска и неопределенности, т.к. каждая альтернатива будет проанализирована и вероятность выбора худшей из них крайне мала. Недостатком является громоздкость построения иерархии.

1.4 Анализ методов критериального анализа

При решении задачи выбора поставщика для ресторанной компании редко случается такое, что имеется четкая информация о том «Что? Сколько? Когда? И как?». Гораздо чаще приходится иметь дело с довольно абстрактными сведениями, т.к. доходы и расходы заведения постоянно варьируются. Кроме того критерии, которые могут интересовать ЛПР могут иметь разные единицы измерения (например, время доставки товара и его цена), и их может быть огромное количество. Также некоторые методы предполагают довольно объемные вычисления.

Исходя из условий, представленных выше, можно сделать вывод, что необходимый метод должен предполагать работу с большим количеством критериев, имеющих разные единицы измерения, и осуществлять максимально простые и минимальные вычисления.

В качестве анализа методов критериального анализа была составлена таблица, в которой сравниваются рассмотренные методы по выявленным ранее критериям.

Таблица 1 – Сравнение методов критериального анализа

Критерии Метод	Работа с нечеткой информацией	Работа с критериями, имеющими разные единицы измерения и значимость	Работа с большим количеством критериев	Простота вычислений	Итого
Метод равномерной оптимизации	-	-	+	+	2
Метод справедливого компромисса	-	-	+	+	2
Метод свертывания критериев	-	+	-	-	1
Метод главного критерия	-	-	-	-	0
Метод идеальной точки	-	+	-	-	1
Аксиоматический метод	+	+	+	-	3
Прямые методы	+	+	-	-	2
Метод компенсации	+	+	-	-	2
Метод анализа иерархий	+	+	+	+	4

В случае выбора поставщика вероятнее всего будет проводиться выбор в условиях неопределенности и риска, так как на изменение обстоятельств может повлиять огромное количество факторов, например, повышение цен, непредвиденные расходы и т.д.

Проанализировав методы критериального анализа, для решения поставленной задачи был выбран метод анализа иерархий, т.к. имеется возможность работы с нечеткой информацией, не имеет значение единица измерения критерия, можно работать с большим количеством критериев, показатели которых непостоянны, математические вычисления довольно просты. Единственным минусом выступает громоздкость иерархии при большом количестве критериев.

1.5 Алгоритмическое решение задачи выбора поставщика для ресторанной компании

Используя для решения поставленной задачи метод анализа иерархий, должны быть выполнены следующие этапы:

1. Определение проблемы и типа искомого решения.
2. Структуризация иерархии решений сверху (цель решения), через промежуточные уровни (критерии, от которых зависят последующие элементы), до самого низкого уровня (который обычно представляет собой набор альтернатив).
3. Построение наборов парных матриц сравнения. Каждый элемент с верхнего уровня используется для сравнения с элементом с уровня ниже.
4. Использование приоритетов, полученных из сравнений, для взвешивания приоритетов для уровня, расположенного ниже. Проработать это с каждым элементом. Затем для каждого элемента с уровня ниже добавить его взвешенные значения и получить общий или глобальный приоритет. Продолжать этот процесс взвешивания и добавления до тех пор, пока не будут получены окончательные приоритеты альтернатив на самом нижнем уровне [6].

Исходя из этапов, описанных выше, алгоритм нахождения оптимального решения с помощью метода анализа иерархий выглядит следующим образом:

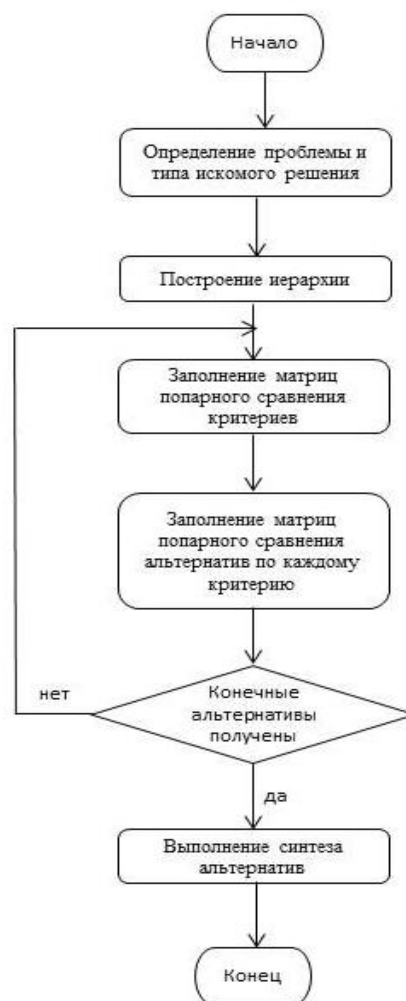


Рисунок 1 – Алгоритм решения задачи многокритериального анализа нечеткой информации

Чтобы произвести сравнение альтернатив, необходима шкала чисел, которая показывает, как сильно один элемент доминирует над другим элементом по отношению к критерию или свойству, по которому они сравниваются. Также эта шкала называется шкалой интенсивности.

Применим данный алгоритм к решению задачи выбора поставщика для ресторанной компании.

Вывод по главе 1

В данной главе была сформулирована задача выбора поставщика, рассмотрены методы для ее решения и выбран тот, который для этого является наиболее подходящим.

2 РЕШЕНИЕ ЗАДАЧИ КРИТЕРИАЛЬНОГО АНАЛИЗА О ВЫБОРЕ ПОСТАВЩИКА ДЛЯ РЕСТОРАННОЙ КОМПАНИИ

2.1 Решение задачи о выборе среди нескольких альтернатив методом анализа иерархий

Пусть имеется некоторое количество альтернатив A_n , где $n = 1, 2, \dots$. Также имеется некоторое количество критериев K_n , где $n = 1, 2, \dots$, по которым эти альтернативы оцениваются. Необходимо провести оценку критериев относительно цели и альтернатив относительно критериев, учитывая предпочтения ЛПР. Оценка производится по шкале, представленной в таблице 2. Данные записываются в виде квадратной матрицы размерности $n \times n$. В общем виде матрица оценивания критериев относительно цели и матрица сравнения альтернатив относительно критерия представлены в таблицах 3 и 4 соответственно, где a_{ij} – оценка предпочтений ЛПР.

Таблица 2 – Шкала интенсивности

Интенсивность важности	Определение	Примечание
1	Равная важность.	Два критерия в равной степени способствуют достижению цели.
2	Слабый или незначительный.	
3-4	Умеренное превосходство.	Опыт и суждение одного критерия немного доминируют над другим.
5-6	Значительное превосходство.	Опыт и суждение одного критерия значительно доминируют над другим.
7-8	Очень сильное превосходство или показательное значение.	Один критерий сильно зависит от другого; его доминирование продемонстрировано на практике.
9	Максимальное превосходство.	Доказательства, которые способствуют доминированию одного критерия над другим, имеют наивысший возможный порядок утверждения.

Таблица 3 – Матрица сравнения критериев относительно цели

Цель	K_1	K_2	...	K_n
K_1	a_{11}	a_{12}	...	a_{1n}
K_2	a_{21}	a_{22}	...	a_{2n}
...
K_n	a_{n1}	a_{n2}	...	a_{nn}

Таблица 4 – Матрица сравнения альтернатив относительно критерия

K_n	A_1	A_2	...	A_n
A_1	a_{11}	a_{12}	...	a_{1n}
A_2	a_{21}	a_{22}	...	a_{2n}
...
A_n	a_{n1}	a_{n2}	...	a_{nn}

Далее необходимо по данной формуле вычислить геометрическое среднее для каждой строки матриц, представленных в таблицах 3 и 4.

$$\omega_i = \sqrt[n]{\prod_{j=1}^n a_{ij}} \quad (7)$$

где a_{ij} – элемент матрицы, являющийся оценкой;

n – размерность матрицы;

$j = 1, n$.

Далее находится нормирующий коэффициент для каждой матрицы:

$$r = \sum_{i=1}^n \omega_i \quad (8)$$

где ω_i – ранее найденное геометрическое среднее.

И в последнюю очередь вычисляется вектор приоритетов для каждой матрицы:

$$q_i = \frac{\omega_i}{r} \quad (9)$$

где r – нормирующий коэффициент;

ω_i – геометрическое среднее;

$i = 1, n$.

Полученный вектор добавляется в матрицу попарного сравнения критериев и альтернатив. В результате чего матрицы приобретают следующий вид, представленный в таблицах 5 и 6.

Таблица 5 – Матрица сравнения критериев относительно цели

Цель	K_1	K_2	...	K_n	q_i
K_1	a_{11}	a_{12}	...	a_{1n}	q_{1i}
K_2	a_{21}	a_{22}	...	a_{2n}	q_{2i}
...
K_n	a_{n1}	a_{n2}	...	a_{nn}	q_{ni}

Таблица 6 – Матрица сравнения альтернатив относительно критерия

K_n	A_1	A_2	...	A_n	q_i
A_1	a_{11}	a_{12}	...	a_{1n}	q_{1i}
A_2	a_{21}	a_{22}	...	a_{2n}	q_{2i}
...
A_n	a_{n1}	a_{n2}	...	a_{nn}	q_{ni}

Далее в несколько этапов определяется согласованность локальных приоритетов для каждой из матриц, рассмотренных выше. На первом этапе определяется сумма каждого j -го столбца матрицы, для которой находится данный показатель:

$$s_i = \sum_{j=1}^n a_{ij} \quad (10)$$

где a_{ij} – элемент матрицы;

$j = 1, n$ – номер столбца матрицы.

Затем полученный результат умножается на i -ю компоненту нормализованного вектора приоритетов q_i :

$$p_i = s_i \cdot q_i \quad (11)$$

где

s_i – сумма каждого j -го столбца матрицы;

q_i – вектор приоритетов;

$i = 1, n$.

Сумма чисел p_i отражает пропорциональность предпочтений λ_{max} , чем ближе эта величина к n (числу объектов и видов действия в матрице парных сравнений), тем более согласованны суждения [6]:

$$\lambda_{max} = \sum_{i=1}^n p_i \quad (12)$$

Отклонение от согласованности выражается индексом согласованности (ИС):

$$ИС = \frac{\lambda_{max} - n}{n - 1} \quad (13)$$

где n – размерность матрицы;

λ_{max} – пропорциональность предпочтений.

Отношение ИС к среднему значению случайного индекса согласованности (СИ) называется отношением согласованности (ОС), где СИ табличная величина.

Таблица 7 – Средние значения случайного индекса согласованности [14]

Размер матрицы, n	1	2	3	4	5	6	7	8	9	10
СИ	0	0	0.58	0.9	1.12	1.24	1.32	1.4	1.45	1.49

Формула нахождения отношения согласованности имеет следующий вид:

$$ОС = \frac{ИС}{СИ} \quad (14)$$

В завершение, осуществляется непосредственно синтез альтернатив для каждой альтернативы, которые будут представлены в условии задачи. Вычисляются приоритеты альтернатив согласно формуле:

$$Q_i = \prod_{j=1}^n q_i \cdot q_j \quad (15)$$

где q_i – вектор приоритета из матрицы сравнения критериев;

q_j – вектор приоритета из матриц сравнения альтернатив.

Среди полученных значений выбирается наибольшее, и именно оно является оптимальной альтернативой для лица, принимающего решение.

2.2 Решение задачи о выборе среди нескольких альтернатив посредством математического анализа

Рассмотрим пример работы алгоритма МАИ. Пусть имеется три поставщика для ресторанной компании, поставляющих чай. Они обладают определенными характеристиками, а именно: цена на товар, качество товара, минимальный объем заказа, время поставки и разнообразие ассортимента. В ходе решения поставленной задачи необходимо сделать оптимальный выбор поставщика.

Требуется сделать выбор из трех предлагаемых поставщиков:

- «ПрофиШеф». Компания находится в г. Тольятти, а значит, доставка будет осуществляться быстро, компания имеет огромный ассортимент, минимальный объем заказа не имеет значения, но цены довольно высокие [9]. Согласно отзывам о компании, можно предположить, что продукция является качественной [11].

- «Русская чайная компания». Компания располагается в г. Москва, следовательно, доставка займет какое-то время, у компании довольно большой ассортимент, но в отличие от первого поставщика у них отсутствуют добавки в чай. Цены на порядок ниже, чем у первого поставщика. Минимальный объем

заказа значения не имеет [12]. Согласно отзывам можно предположить, что товар имеет среднее качество [16].

- «Tea Point». Компания находится в г. Владивосток, следовательно, доставка будет осуществляться дольше, чем у предыдущих поставщиков. Ассортимент меньше, чем у компаний, представленных выше. Цены средние, но выше, чем у второго поставщика. Минимальный объем заказа не имеет значения [13]. Согласно отзывам, можно предположить, что продукция высокого качества [19].

На первом этапе МАИ осуществляется представление проблемы в виде иерархии. Имеется три поставщика:

- A_1 – «ПрофиШеф»;
- A_2 – «Русская чайная компания»;
- A_3 – «Tea Point».

Они составляют третий низший уровень иерархии, и являются альтернативами.

Критериями, по отношению к которым оцениваются альтернативы, являются:

- K_1 – время доставки товара;
- K_2 – разнообразие ассортимента;
- K_3 – цена товара;
- K_4 – минимальный объем заказа;
- K_5 – качество товара.

Построенная иерархия представлена на рисунке 2.

На втором этапе необходимо установить приоритеты критериев и произвести оценку каждой из альтернатив по критериям [5].

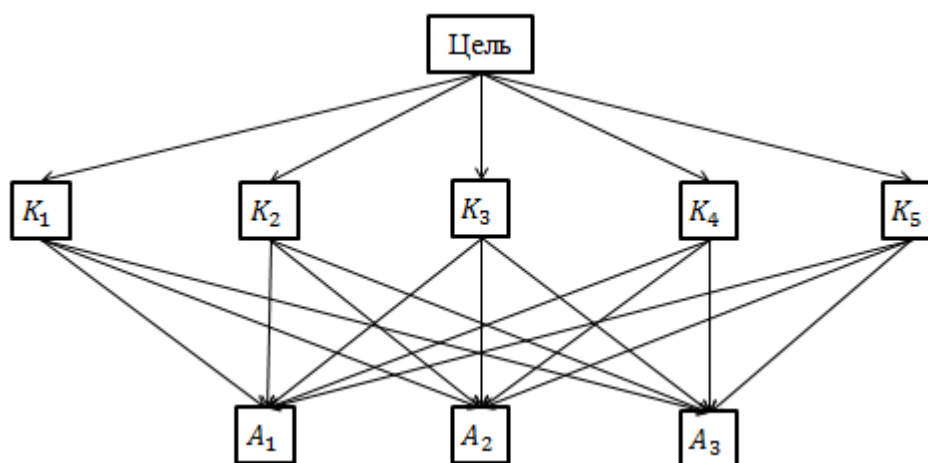


Рисунок 2 – Иерархия поставленной задачи

Составим матрицу для попарного сравнения критериев на втором уровне по отношению к общей цели, расположенной на первом уровне. Для этого будут использованы данные таблицы 2. Полученные сведения будут занесены в таблицу 8. Матрицы попарного сравнения каждой альтернативы на третьем уровне по отношению к критериям второго уровня будут соответственно занесены в таблицы 9-13. Данные матрицы, как правило, составляются на основе предпочтений ЛПР.

Таблица 8 – Матрица попарного сравнения критериев

Цель	K_1	K_2	K_3	K_4	K_5
K_1	1	3	1/6	1/7	1/6
K_2	1/3	1	1/5	1/4	1/6
K_3	6	5	1	1/5	3
K_4	7	4	5	1	1
K_5	6	6	1/3	1	1

Таблица 9 – Матрица попарного сравнения альтернатив по отношению к K_1

K_1	A_1	A_2	A_3
A_1	1	7	9
A_2	1/7	1	7
A_3	1/9	1/7	1

Таблица 10 – Матрица попарного сравнения альтернатив по отношению к K_2

K_2	A_1	A_2	A_3
A_1	1	5	2
A_2	1/5	1	7
A_3	1/2	1/7	1

Таблица 11 – Матрица попарного сравнения альтернатив по отношению к K_3

K_3	A_1	A_2	A_3
A_1	1	1/7	1/5
A_2	7	1	5
A_3	5	1/5	1

Таблица 12 – Матрица попарного сравнения альтернатив по отношению к K_4

K_4	A_1	A_2	A_3
A_1	1	1	1
A_2	1	1	1
A_3	1	1	1

Таблица 13 – Матрица попарного сравнения альтернатив по отношению к K_5

K_5	A_1	A_2	A_3
A_1	1	7	1
A_2	1/7	1	7
A_3	1	1/7	1

На третьем этапе осуществляется синтез приоритетов. Для этого необходимо найти геометрическое среднее каждой строки матрицы, нормирующий вектор и затем вектор приоритетов.

Размерность матрицы попарного сравнения критериев, представленной в таблице 3, равна $n = 5$. Соответственно, геометрическое среднее для каждой строки этой матрицы вычисляется по формуле (7). В результате имеем:

$$\omega_1 = \sqrt[n]{a_{11} \cdot a_{12} \cdot a_{13} \cdot a_{14} \cdot a_{15}} = \sqrt[5]{1 \cdot 3 \cdot \frac{1}{6} \cdot \frac{1}{7} \cdot \frac{1}{6}} = \sqrt[5]{0.012} \approx 0.412$$

$$\omega_2 = \sqrt[n]{a_{21} \cdot a_{22} \cdot a_{23} \cdot a_{24} \cdot a_{25}} = \sqrt[5]{\frac{1}{3} \cdot 1 \cdot \frac{1}{5} \cdot \frac{1}{4} \cdot \frac{1}{6}} = \sqrt[5]{0.0027} \approx 0.308$$

$$\omega_3 = \sqrt[n]{a_{31} \cdot a_{32} \cdot a_{33} \cdot a_{34} \cdot a_{35}} = \sqrt[5]{6 \cdot 5 \cdot 1 \cdot \frac{1}{5} \cdot 3} = \sqrt[5]{18} \approx 1.782$$

$$\omega_4 = \sqrt[n]{a_{41} \cdot a_{42} \cdot a_{43} \cdot a_{44} \cdot a_{45}} = \sqrt[5]{7 \cdot 4 \cdot 5 \cdot 1 \cdot 1} = \sqrt[5]{140} \approx 2.686$$

$$\omega_5 = \sqrt[n]{a_{51} \cdot a_{52} \cdot a_{53} \cdot a_{54} \cdot a_{55}} = \sqrt[5]{6 \cdot 6 \cdot \frac{1}{3} \cdot 1 \cdot 1} = \sqrt[5]{12} \approx 1.643$$

Далее находится нормирующий коэффициент согласно формуле (8). Соответственно:

$$r = 0.412 + 0.308 + 1.782 + 2.686 + 1.643 = 6.831$$

И в последнюю очередь вычисляется вектор приоритетов матрицы, представленной в таблице 3, согласно формуле (9):

$$q_1 = \frac{\omega_1}{r} = 0.412/6.831 \approx 0.0603$$

$$q_2 = \frac{\omega_2}{r} = 0.308/6.831 \approx 0.045$$

$$q_3 = \frac{\omega_3}{r} = 1.782/6.831 \approx 0.2608$$

$$q_4 = \frac{\omega_4}{r} = 2.686/6.831 \approx 0.393$$

$$q_5 = \frac{\omega_5}{r} = 1.643/6.831 \approx 0.2405$$

Полученный вектор добавляется в матрицу попарного сравнения критериев, результат представлен в таблице 14.

Таблица 14 – Матрица попарного сравнения критериев

Цель	K_1	K_2	K_3	K_4	K_5	q_1
K_1	1	3	1/6	1/7	1/6	0.0603
K_2	1/3	1	1/5	1/4	1/6	0.045
K_3	6	5	1	1/5	3	0.2608
K_4	7	4	5	1	1	0.393
K_5	6	6	1/3	1	1	0.2405

Аналогичную процедуру осуществляем для всех матриц попарных сравнений. Результаты представлены в таблицах 15-19.

Таблица 15 – Матрица попарного сравнения альтернатив по отношению к K_1

K_1	A_1	A_2	A_3	q_2
A_1	1	7	9	1.784
K_1	A_1	A_2	A_3	q_2
A_2	1/7	1	7	0.448
A_3	1/9	1/7	1	0.112

Таблица 16 – Матрица попарного сравнения альтернатив по отношению к K_2

K_2	A_1	A_2	A_3	q_3
A_1	1	5	2	0.584
A_2	1/5	1	7	0.303
A_3	1/2	1/7	1	0.112

Таблица 17 – Матрица попарного сравнения альтернатив по отношению к K_3

K_3	A_1	A_2	A_3	q_4
A_1	1	1/7	1/5	0.066
A_2	7	1	5	0.714
A_3	5	1/5	1	0.218

Таблица 18 – Матрица попарного сравнения альтернатив по отношению к K_4

K_4	A_1	A_2	A_3	q_5
A_1	1	1	1	0.333
A_2	1	1	1	0.333
A_3	1	1	1	0.333

Таблица 19 – Матрица попарного сравнения альтернатив по отношению к K_5

K_5	A_1	A_2	A_3	q_6
A_1	1	7	1	0.556
A_2	1/7	1	7	0.291
A_3	1	1/7	1	0.152

Далее определяется согласованность локальных приоритетов для каждой из матриц, рассмотренных выше. Найдем отношение согласованности (ОС) для матрицы попарного сравнения критериев, представленных в таблице 3.

На первом этапе определяется сумма каждого j -го столбца матрицы согласно формуле (10):

$$s_1 = 1 + \frac{1}{3} + 6 + 7 + 6 = 20.333$$

$$s_2 = 3 + 1 + 5 + 4 + 6 = 19$$

$$s_3 = \frac{1}{6} + \frac{1}{5} + 1 + 5 + \frac{1}{3} = 6.7$$

$$s_4 = \frac{1}{7} + \frac{1}{4} + \frac{1}{5} + 1 + 1 = 2.592$$

$$s_5 = \frac{1}{6} + \frac{1}{6} + 3 + 1 + 1 = 5.333$$

Затем полученный результат умножается на i -ю компоненту нормализованного вектора приоритетов q согласно формуле (11). Отсюда следует:

$$p_1 = s_1 \cdot q_{11} = 20.333 \cdot 0.0603 = 1.226$$

$$p_2 = s_2 \cdot q_{21} = 19 \cdot 0.045 = 0.855$$

$$p_3 = s_3 \cdot q_{31} = 6.7 \cdot 0.2608 = 1.747$$

$$p_4 = s_4 \cdot q_{41} = 2.592 \cdot 0.393 = 1.018$$

$$p_5 = s_5 \cdot q_{51} = 5.333 \cdot 0.2405 = 1.282$$

Далее найдем согласованность суждений по формуле (12):

$$\begin{aligned} \lambda_{max} &= p_1 + p_2 + p_3 + p_4 + p_5 = 1.226 + 0.855 + 1.747 + 1.018 + 1.282 = \\ &= 6.128 \end{aligned}$$

Затем найдем индекс согласованности согласно формуле (13):

$$ИС = \frac{6.128 - 5}{5 - 1} = 0.282$$

После необходимо найти отношение согласованности по формуле (14).

Отсюда следует:

$$ОС = \frac{0.282}{1.12} = 0.251$$

Аналогичным образом вычислим ОС для матриц, представленных в таблицах 9-13. В результате проделанных вычислений имеем:

- для K_1 : ОС = 4,125;
- для K_2 : ОС = 0,837;
- для K_3 : ОС = 0,143;
- для K_4 : ОС = 0;
- для K_5 : ОС = 1,661.

В завершение, осуществляется непосредственно синтез альтернатив, вычисляются приоритеты всех трех альтернатив по формуле (15):

$$Q_1 = 1.784 * 0.0603 + 0.584 * 0.045 + 0.066 * 0.2608 + 0.333 * 0.393 + \\ + 0.556 * 0.2405 = 0.415$$

$$Q_2 = 0.448 * 0.0603 + 0.303 * 0.045 + 0.714 * 0.2608 + 0.333 * 0.393 + \\ + 0.291 * 0.2405 = 0.427$$

$$Q_3 = 0.112 * 0.0603 + 0.112 * 0.045 + 0.218 * 0.2608 + 0.333 * 0.393 + \\ + 0.152 * 0.2405 = 0.236$$

Данные приоритеты, полученные в результате синтеза альтернатив, являются показателями оптимальности. Оптимальным выбором является тот, у

которого наибольший приоритет Q_n . Приоритет у альтернативы A_2 , которой является поставщик «Русская чайная компания», выше, чем у всех остальных. Таким образом, выбор поставщика «Русская чайная компания» является оптимальным для заданного ЛПР.

2.3 Создание программы, реализующей математическую модель метода анализа иерархий

Создаваемое приложение должно осуществлять решение задачи многокритериального анализа посредством МАИ. МАИ работает на основе оценок, которые предоставляет лицо, принимающее решение. Следовательно, они могут варьироваться в зависимости от предпочтений ЛПР и прописывать их в качестве оценок по умолчанию было бы не логично. Следовательно, необходим ввод оценок критериев и оценок альтернатив вручную. Оценка критериев вводится в таблицу, в которой элементы главной диагонали неизменно равны единице. Также для удобства пользователя и во избежание ошибки ввода оценок на панели должна присутствовать некоторая шкала оценки, обеспечивающая корректный ввод данных. Промежуточные значения рассчитываются динамически, сразу после ввода пользователем оценок. Также имеются кнопки, осуществляющие переход между оцениванием различных факторов такие как:

- цель (осуществляет переход на окно, в котором происходит оценка критериев относительно цели);
- время доставки товара (осуществляет переход на окно, в котором происходит оценка альтернатив относительно критерия «Время доставки товара»);
- разнообразие ассортимента (осуществляет переход на окно, в котором происходит оценка альтернатив относительно критерия «Разнообразие ассортимента»);

- цена товара (осуществляет переход на окно, в котором происходит оценка альтернатив относительно критерия «Цена товара»);
- минимальный объем заказа (осуществляет переход на окно, в котором происходит оценка альтернатив относительно критерия «Минимальный объем заказа»);
- качество товара (осуществляет переход на окно, в котором происходит оценка альтернатив относительно критерия «Качество товара»).

И в завершение необходим список факторов, которые оценивает пользователь. При нажатии на кнопку «ОК» выводится некоторая диаграмма, на которой наглядно показано, какая из альтернатив является более предпочтительной.

Исходя из требований, представленных выше, программа реализована на языке Java с использованием технологии JavaFX, которая позволяет максимально легко создавать графический интерфейс. Полный код программы представлен в приложении А. В результате работы была разработана программа, снимки работы которой представлены на рисунке 3-5.

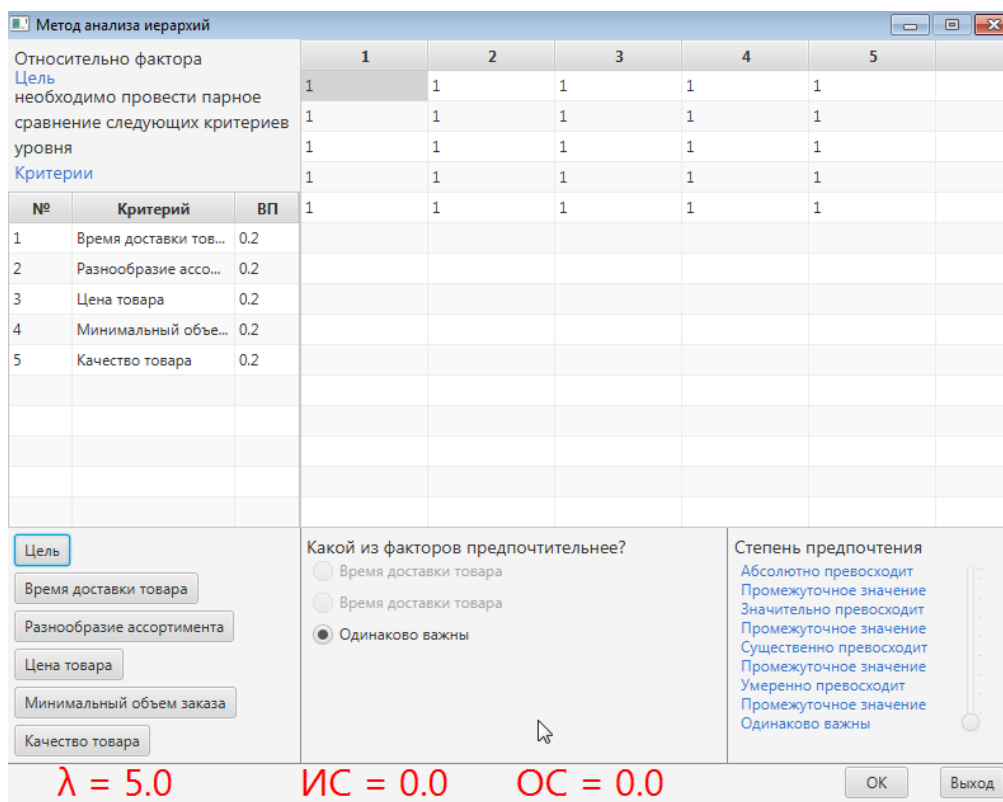


Рисунок 3 – Снимок работы программы в результате нажатия на кнопку «Цель»

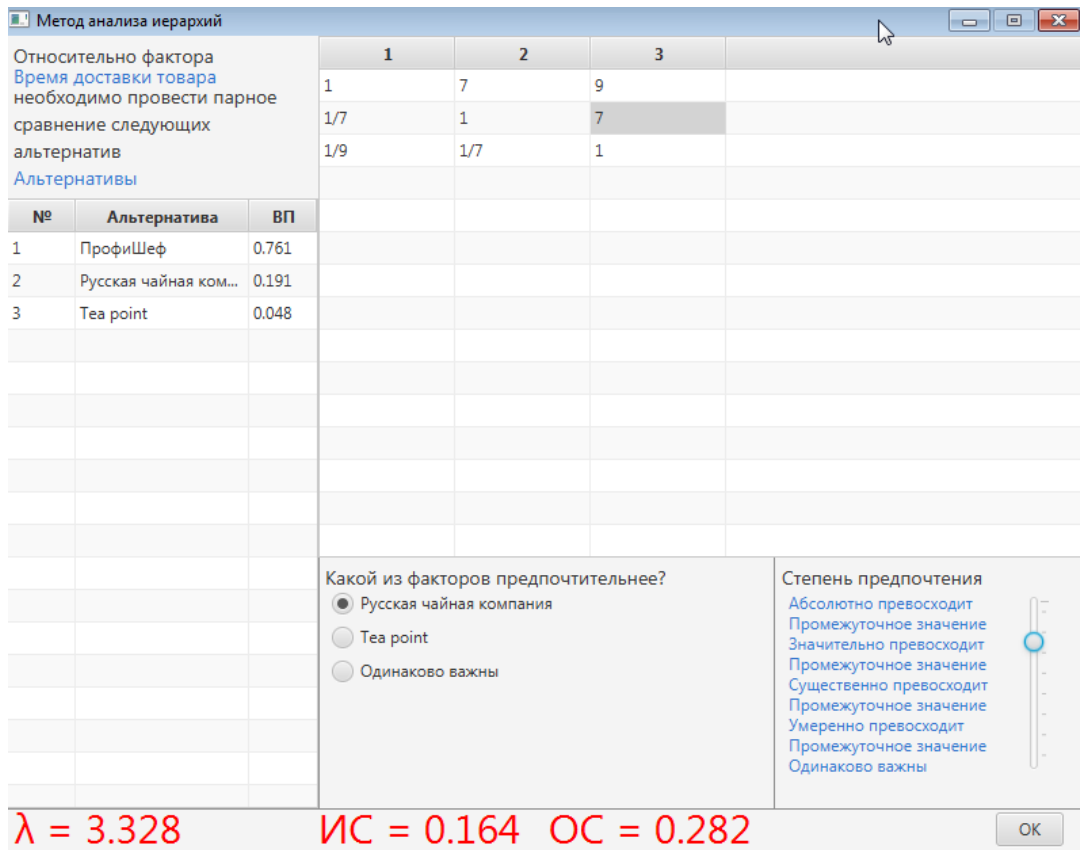


Рисунок 4 – Снимок работы программы «Время доставки товара»

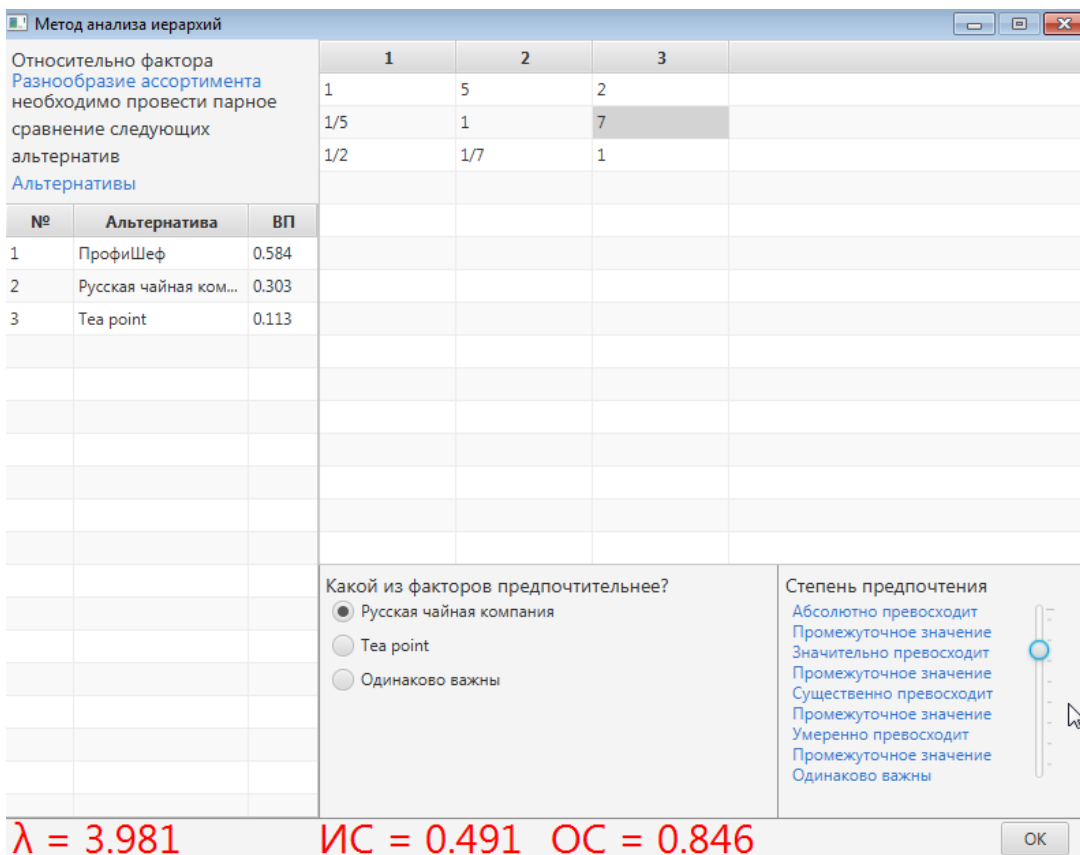


Рисунок 5 – Снимок работы программы «Разнообразие ассортимента»

В результате созданная программа позволяет ввести оценки, являющиеся, по сути, предпочтениями ЛПР, с помощью шкалы «степень предпочтения», которая в свою очередь исключает возможность ввода числа за пределами доступного диапазона [1; 9], динамически осуществляет расчеты и в завершение предоставляет круговую диаграмму, демонстрирующую результаты вычислений.

2.4 Проверка правильности работы программы, реализующей метод анализа иерархий

В ходе проверки правильности работы программы, были введены те же значения, которые использовались при реализации метода анализа иерархий посредством математического анализа вручную. В результате работы программы была получена диаграмма, которая представлена на рисунке 4.

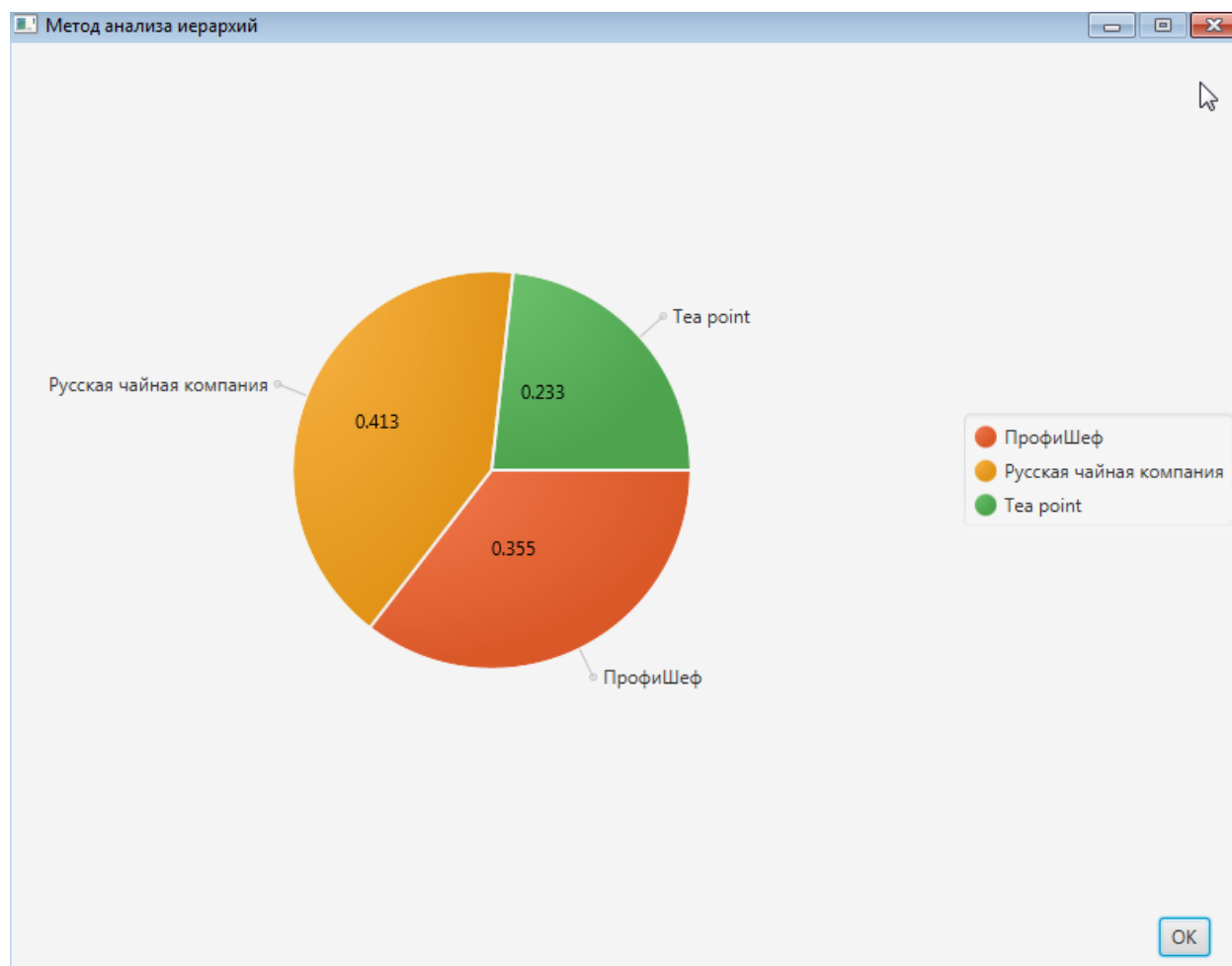


Рисунок 6 – Диаграмма, полученная в результате работы программы

Как видно из рисунка «Русская чайная компания» является более предпочтительной альтернативой относительно двух других. Далее следует компания «ПрофиШеф» и в завершение «TeaPoint». Аналогичный результат был получен в ходе вычислений вручную в пункте 2.1. Имеются небольшие различия в числовых значениях. Это связано с тем, что в ходе вычислений вручную производилось округление чисел до четвертого знака после запятой, в отличие от программы, которая проводила вычисления максимально точно.

Вывод по главе 2

В данной главе была составлена математическая модель метода анализа иерархий, приведен пример его работы на основе выбора поставщика чайной продукции, а также представлена программа, реализующая данную математическую модель.

3 ОПРЕДЕЛЕНИЕ ОПТИМАЛЬНОГО РЕШЕНИЯ В ВЫБОРЕ ПОСТАВЩИКА С ИСПОЛЬЗОВАНИЕМ МЕТОДА АНАЛИЗА ИЕРАРХИЙ

3.1 Исходные данные для задачи о выборе поставщика

Исходный алгоритм метода анализа иерархий предполагает, что все товары будут куплены у одного поставщика. Однако, в жизни такое маловероятно, и скорее всего товары будут куплены у разных поставщиков. Задача заключается в том, что пользователю необходимо предоставить альтернативы не только при покупке товаров у одного поставщика, но и в случае, если товары будут приобретены у разных поставщиков. В таком случае выбор среди этих альтернатив будет сделан в пользу той, которая имеет наибольший приоритет.

Пусть имеется некоторый пакет товаров, которые необходимо купить: кофе, мед, чай. Также имеется три поставщика, которые были ранее представлены в пункте 2.1: «ПрофиШеф», «Русская чайная компания», «TeaPoint». Данные поставщики предлагают следующие товары:

- «ПрофиШеф» - чай, топпинг, сироп, фруктовая основа, кофе;
- «Русская чайная компания» - чай, кофе, посуда, мед, шоколад;
- «TeaPoint» - чай, кофе, посуда, шоколад, подарочные наборы.

При решении задачи будет сохраняться изначальный алгоритм метода анализа иерархий. Отличие заключается в том, что каждый продукт будет оцениваться отдельно, что позволит выяснить какая альтернатива является более выгодной [18]: покупать продукты отдельно или же приобрести весь пакет у одного поставщика.

3.2 Решение задачи выбора поставщика с использованием метода анализа иерархий

Глядя на пакет, представленный в пункте 3.1, который необходимо приобрести, можно сразу сказать, что мед будет приобретен у поставщика

«Русская чайная компания», так как прочие поставщики этот товар не предоставляют. Соответственно, данная альтернатива будет иметь наивысший приоритет равный единице. Следовательно, будет производиться оценка чая и кофе.

Альтернативы и критерии те же, что и в пункте 2.1. Имеются поставщики, которые также являются альтернативами:

- A_1 – «ПрофиШеф»;
- A_2 – «Русская чайная компания»;
- A_3 – «Tea Point».

Также имеются критерии, по отношению к которым оцениваются альтернативы:

- K_1 – время доставки товара;
- K_2 – разнообразие ассортимента;
- K_3 – цена товара;
- K_4 – минимальный объем заказа;
- K_5 – качество товара.

Оценки критериев относительно цели взяты из таблицы 14. Также сохранены оценки критериев «время доставки товара», «минимальный объем заказа», представленные в таблицах 15 и 18 соответственно, так как они остаются неизменными вне зависимости от товара. Оценка критерия «качество товара», представленная в таблице 19, была произведена, опираясь на отзывы в интернете, поэтому она также остается неизменной. Далее в таблицах 20-21 представлены оценки альтернатив относительно товара «кофе».

Таблица 20 – Матрица попарного сравнения альтернатив по отношению к K_2

K_1	A_1	A_2	A_3
A_1	1	1/9	1/6
A_2	9	1	7
A_3	6	1/7	1

Таблица 21 – Матрица попарного сравнения альтернатив по отношению к K_3

K_2	A_1	A_2	A_3
A_1	1	4	1/7
A_2	1/4	1	6
K_2	A_1	A_2	A_3
A_3	7	1/6	1

Осуществив те же действия с матрицами, что и в пункте 2.1 имеем следующие значения альтернатив:

- $q_1 = 0.331$;
- $q_2 = 0.356$;
- $q_3 = 0.313$.

Отсюда следует, что кофе лучше покупать у поставщика «Русская чайная компания».

Проведем аналогичные действия относительно товара «чай». Оценки представлены в таблицах 22 и 23.

Таблица 22 – Матрица попарного сравнения альтернатив по отношению к K_2

K_1	A_1	A_2	A_3
A_1	1	1/8	6
A_2	8	1	9
A_3	1/6	1/9	1

Таблица 23 – Матрица попарного сравнения альтернатив по отношению к K_3

K_2	A_1	A_2	A_3
A_1	1	1/5	1/9
A_2	5	1	1/6
A_3	9	6	1

В ходе последующих вычислений были получены следующие значения альтернатив:

- $q_1 = 0.279$;

- $q_2 = 0.307$;
- $q_3 = 0.413$.

Отсюда следует вывод, что кофе лучше приобретать у поставщика «TeaPoint».

Далее необходимо найти среднее геометрическое от альтернатив по каждому товару и сравнить с наиболее оптимальной альтернативой, полученной в пункте 2.1.

$$Q_1 = \sqrt[3]{1 + 0,356 + 0,413} = 1,209$$

$$Q_2 = 0.413$$

где Q_1 – альтернатива, полученная в ходе покупки товара у разных поставщиков;

Q_2 – оптимальная альтернатива, полученная в пункте 2.1.

В результате полученные значения существенно различаются. Отсюда можно сделать однозначный вывод, что покупать товар у разных поставщиков выгоднее, чем брать весь пакет у одного поставщика.

Исходя из того, что все расчеты производятся на основе оценок ЛПР, можно сделать вывод, что как бы они не менялись, разработанная математическая модель выявит именно ту альтернативу, которая и будет являться оптимальной для ЛПР. Как было сказано ранее оптимальным выбором является тот, у которого наибольший приоритет Q_n .

Вывод по главе 3

В данной главе был применен метод анализа иерархий в случае выбора среди нескольких поставщиков одновременно и показана эффективность данного подхода.

ЗАКЛЮЧЕНИЕ

Принятие важных решений является неотъемлемой частью не только крупных компаний, но и обычных людей.

В данной бакалаврской работе был выделен объект исследования, проанализирована предметная область, определены цели и задачи работы. Проведен анализ существующих методов многокритериального анализа и выявлен наиболее подходящий для реализации цели выпускной квалификационной работы.

При анализе МКА было выявлено, что для решения задач бакалаврской работы потребуется написать свою программную реализацию алгоритма метода анализа иерархий, а не использовать существующие, так как существующие разработки были сделаны для других конкретных целей, поэтому и наша разработка будет носить такой же характер.

Был спроектирован интерфейс пользователя и использован алгоритм МАИ, а также реализовано визуальное представление оптимальности альтернатив.

Далее алгоритмы и интерфейс были реализованы в среде разработки IntelliJ IDEA с использованием языка программирования Java. В ходе выполнения работы реализованы следующие функции:

- ввод пользователем оценок, отражающих его предпочтения;
- отражение промежуточных расчетов;
- реализовано визуальное представление оптимальности альтернатив.

Затем, алгоритм МАИ был применен, с учетом того, что товары приобретаются у разных поставщиков, так как данное явление очень часто встречается в реальной жизни.

СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

Научная и методическая литература

1. Семенов С.С. Обзор методов принятия решений при разработке сложных технических систем [Текст] // Функциональная надежность. Теория и практика. / Полтавский А.В., Маклаков В.В., Крянев А.В. – Надежность, 2014. – С. 73-74.
2. Беляев И.П. Многокритериальные методы принятия решений [Текст]– С. 16-17.
3. Борисов А.Н. Обработка нечеткой информации в системах принятия решений [Текст] / Алексеев А.В., Меркурьева Г.В., Сладзь Н.Н., Глушков В.И. – М.: Радио и связь, 2003 – С. 235-237.
4. Прохоров Ю.К. Управленческие решения: Учебное пособие.[Текст] / Прохоров Ю.К., Фролов В. В. – 2-е изд., испр. и доп. – СПб: СПбГУ ИТМО, 2011. – 138 с.
5. Молодой ученый [Текст] : междунар. научн. журн / учредитель ООО «Издательство Молодой ученый». – 2016, февраль – Казань. Издательство Молодой ученый. – двухмес. – ISSN 2072-0297. 2016, №3 (107) – 500 экз.
6. Беляев И.П. Принцип инвариантности лидерства [Текст] – С. 1.
7. Научные ведомости [Текст] : научн. реценз. журн. / учредитель ФГАОУ ВПО «Белгородский государственный национальный исследовательский университет». – 2014, январь – Белгород. Белгородский государственный национальный исследовательский университет – трехмес. – ISSN 2411-3808. 2014, №1 (172).
8. Молодежный научно-технический вестник [Текст] : электрон. журн. / учредитель ООО «Академия инженерных наук им. А.М. Прохорова». – 2014, ноябрь – М. Академия инженерных наук им. А.М. Прохорова. – ежемес. – ISSN 2307-0609. 2014, №11.

Электронные ресурсы

9. NooTron [Электронный ресурс] // Часть № 2 Многокритериальный анализ: методы и системные задачи. URL: <http://nootron.net.ua/Help.htm?p=chapter2.html> (дата обращения 19.12.17);
10. ПрофиШеф [Электронный ресурс] // О компании. URL: <http://profi-chef.ru/about/> (дата обращения: 10.01.2018);
11. Отзовик [Электронный ресурс] // Компания "Профи Шеф" (Россия, Тольятти) – Отзывы URL: http://otzovik.com/reviews/kompaniya_profi_shef_russia_tolyatti/ (дата обращения: 10.01.2018);
12. Русская чайная компания [Электронный ресурс] // Москва. Центральный офис. URL: <https://www.rusteaco.ru/company/contacts/moscow/> (дата обращения: 10.01.2018);
13. СПР [Электронный ресурс] // Отзывы о "русская чайная компания". URL: <https://www.spr.ru/otzyvy/russkaya-chaynaya-kompaniya.html> (дата обращения: 10.01.2018);
14. Tea Point [Электронный ресурс] // О Tea Point URL: <http://tea-point.ru/about-company.php> (дата обращения: 10.01.2018);
15. Справочник компаний [Электронный ресурс] // Tea Point. URL: <https://www.vl.ru/teapoint> (дата обращения: 10.01.2018);

Литература на иностранном языке

16. Patrascu V. Shannon entropy for imprecise and under-defined or over-defined information. [Text] – The 25th Conference on Applied and Industrial Mathematics. – 2017, September. – PP. 85.
17. Longmei L. An Ontology of Preference-Based Multi-objective Metaheuristics. [Text] / Longmei Li, Iryna Yevseyeva, Vitor Basto-Fernandes, Heike Trautmann, Ning Jing, Michael Emmerich – 2017, March. – PP. 1-2.
18. Thomas L. S. Decision making with the analytic hierarchy process. [Text] – Katz Graduate School of Business. – 2008. – PP. 95.

19. Melvin A. Decision-Making using the Analytic Hierarchy Process (AHP) and SAS/IML. [Text] – Social Security Administration, Baltimore. – 2012. – PP. 2.

20. Roy P. A Novel Fuzzy Document Based Information Retrieval Model for Forecasting [Text] / Roy P., Kumar R., Sharma S. – 2017, May. – PP. 138.

ПРИЛОЖЕНИЕ А

Class Main

```
import javafx.application.Application;
import javafx.fxml.FXMLLoader;
import javafx.scene.Parent;
import javafx.scene.Scene;
import javafx.scene.control.TableView;
import javafx.stage.Stage;

public class Main extends Application {

    private static Stage stage;
    private static Scene root;
    private static Criterion c[] = new Criterion[6];
    private static String kr[] = {"Время доставки товара", "Разнообразие
ассортимента", "Цена товара", "Минимальный объем заказа", "Качество товара"};
    private static String kr2[] = {"ПрофиШеф", "Русская чайная компания", "Tea
point"};
    private static double Q1, Q2, Q3;
    private static DiagramController diagramController;

    public static void calcQ(){
        double[] qCriterionK1 = c[1].getqCriterion();
        double[] qCriterionK2 = c[2].getqCriterion();
        double[] qCriterionK3 = c[3].getqCriterion();
        double[] qCriterionK4 = c[4].getqCriterion();
        double[] qCriterionK5 = c[5].getqCriterion();
        double[] qCriterion = c[0].getqCriterion();
        Q1 = qCriterionK1[0] * qCriterion[0] + qCriterionK2[0] * qCriterion[1] +
qCriterionK3[0] * qCriterion[2] + qCriterionK4[0] * qCriterion[3] +
qCriterionK5[0] * qCriterion[4];
        Q2 = qCriterionK1[1] * qCriterion[0] + qCriterionK2[1] * qCriterion[1] +
qCriterionK3[1] * qCriterion[2] + qCriterionK4[1] * qCriterion[3] +
qCriterionK5[1] * qCriterion[4];
        Q3 = qCriterionK1[2] * qCriterion[0] + qCriterionK2[2] * qCriterion[1] +
qCriterionK3[2] * qCriterion[2] + qCriterionK4[2] * qCriterion[3] +
qCriterionK5[2] * qCriterion[4];
    }

    public static double getQ1() {
        return Q1;
    }

    public static double getQ2() {
        return Q2;
    }

    public static double getQ3() {
        return Q3;
    }

    public static Criterion getCriterion(int x){
        return c[x];
    }

    public static String getKr(int i){
        return kr[i];
    }

    public static String getKr2(int i){
        return kr2[i];
    }
}
```



```

    }

    public static void updateDiagram() {
        diagramController.update();
    }

    public void init() throws Exception{
        c[0] = new Criterion(5);
        for(int i = 1; i < 6; i++) c[i] = new Criterion(3);

Parents.setAim(FXMLLoader.load(getClass().getResource("ParentsFXML/aim.fxml")));

Parents.setTime(FXMLLoader.load(getClass().getResource("ParentsFXML/time.fxml"))
);

Parents.setDiff(FXMLLoader.load(getClass().getResource("ParentsFXML/diff.fxml"))
);

Parents.setCost(FXMLLoader.load(getClass().getResource("ParentsFXML/cost.fxml"))
);

Parents.setMin(FXMLLoader.load(getClass().getResource("ParentsFXML/min.fxml")));

Parents.setQuality(FXMLLoader.load(getClass().getResource("ParentsFXML/quality.f
xml")));
        FXMLLoader loader = new
FXMLLoader(getClass().getResource("ParentsFXML/diagram.fxml"));
        Parents.setDiagram(loader.load());
        diagramController = loader.getController();
    }

    @Override
    public void start(Stage primaryStage) throws Exception{
        stage = primaryStage;
        stage.setTitle("Метод анализа иерархий");
        stage.setWidth(800);
        stage.setHeight(630);
        stage.setResizable(false);
        root = new Scene(Parents.getAim());
        stage.setScene(root);
        stage.show();
    }

    public static void changeScene(Parent parent) {
        root.setRoot(parent);
    }

    public static void main(String[] args) {
        launch(args);
    }
}

```

Class Criterion

```

public class Criterion {

    double[][] criterion;
    double[] wCriterion;
    double rCriterion;
    double[] qCriterion;
    double[] sCriterion;
    double[] pCriterion;
    double hmax;
    double IS;
}

```

```

double OS;

public double getQValue(int i){
    return qCriterion[i];
}

public double[][] getCriterion() {
    return criterion;
}

public double[] getqCriterion() {
    return qCriterion;
}

public void change(int i, int j, double val){
    criterion[i][j] = val;
    criterion[j][i] = 1.0 / val;
    calc();
}

public double getValue(int i, int j){
    return criterion[i][j];
}

public double getrCriterion() {
    return rCriterion;
}

public double getHmax() {
    return hmax;
}

public double getIS() {
    return IS;
}

public double getOS() {
    return OS;
}

public Criterion(int size){
    criterion = new double[size][size];
    for(int i = 0; i < size; i++) for(int j = 0; j < size; j++)
criterion[i][j] = 1;
    calc();
}

public void calc(){
    wCriterion = w(criterion);
    rCriterion = r(wCriterion);
    qCriterion = q(wCriterion, rCriterion);
    sCriterion = s(criterion);
    pCriterion = p(sCriterion, qCriterion);
    hmax = hmax(pCriterion);
    IS = IS(hmax, pCriterion);
    OS = OS(pCriterion, IS);
}

private static double[] w(double[][] array) {
    double[] w = new double[array.length];
    for (int i = 0; i < array.length; i++) { //заполняем массив нулями
        for (int j = 0; j < array.length; j++) {
            w[i] = 1;

```

```

    }
}
for (int i = 0; i < array.length; i++) {
    for (int j = 0; j < array.length; j++) {
        w[i] *= Math.pow(array[i][j], 1d / array.length);
    }
}
return w;
}

private static double r(double[] array) {
    double r = 0;
    for (int i = 0; i < array.length; i++) {
        r += array[i];
    }
    return r;
}

private static double[] q(double[] array, double r) {
    double[] q = new double[array.length];
    for (int i = 0; i < array.length; i++) {
        q[i] = array[i] / r;
    }
    return q;
}

private static double[] s(double[][] array) {
    double[] s = new double[array.length];
    for (int j = 0; j < array.length; j++) { //сумма столбцов
        for (int i = 0; i < array.length; i++) {
            s[j] += array[i][j];
        }
    }
    return s;
}

private static double[] p(double[] arrayS, double[] arrayQ) {
    double[] p = new double[arrayS.length];
    for (int i = 0; i < arrayS.length; i++) {
        p[i] = arrayQ[i] * arrayS[i];
    }
    return p;
}

private static double hmax(double[] array) {
    double hmax = 0;
    for (int i = 0; i < array.length; i++) {
        hmax += array[i];
    }
    return hmax;
}

private static double IS(double hmax, double array[]) {
    double IS;
    IS = (hmax - array.length) / (array.length - 1);
    return IS;
}

private static double OS(double[] array, double IS) {
    double OS;
    double SI = 0;
    switch (array.length) {
        case 3:

```

```

        SI = 0.58;
        break;
    case 5:
        SI = 1.12;
        break;
    }
    OS = IS / SI;
    return OS;
}
}

```

Class Controller

```

import javafx.collections.FXCollections;
import javafx.collections.ObservableList;
import javafx.fxml.FXML;
import javafx.scene.Parent;
import javafx.scene.control.*;
import javafx.scene.control.cell.PropertyValueFactory;
import javafx.scene.input.MouseEvent;

public class Controller {

    @FXML
    private TableView table, leftTable;
    @FXML
    private Slider slider;
    @FXML
    private RadioButton rb1, rb2, rb_all;
    @FXML
    private Label lambdaLabel, ISLabel, OSLabel;
    private int cell_i, cell_j;
    private boolean reversed = false;

    private void initTable(){
        TableColumn tc[] = new TableColumn[5];
        for(int i = 0; i < 5; i++){
            tc[i] = new TableColumn(((Integer) (i+1)).toString());
            tc[i].setMinWidth(100);
            tc[i].setMaxWidth(100);
            tc[i].setCellValueFactory(new PropertyValueFactory<>("k" +
((Integer) (i+1)).toString()));
            tc[i].setSortable(false);
        }
        table.getColumns().setAll(tc[0], tc[1], tc[2], tc[3], tc[4]);
        table.getSelectionModel().setSelectionMode(SelectionMode.SINGLE);
        table.getSelectionModel().setCellSelectionEnabled(true);
        table.setItems(getData());
        setLabels();
    }

    private void initLeftTable(){
        TableColumn num = new TableColumn("№");
        num.setMinWidth(50);
        num.setMaxWidth(50);
        num.setCellValueFactory(new PropertyValueFactory<>("num"));
        num.setSortable(false);

        TableColumn name = new TableColumn("Критерий");
        name.setMinWidth(128);
        name.setMaxWidth(128);
        name.setCellValueFactory(new PropertyValueFactory<>("name"));
        name.setSortable(false);
    }
}

```

```

TableColumn vp = new TableColumn("BII");
vp.setMinWidth(50);
vp.setMaxWidth(50);
vp.setCellValueFactory(new PropertyValueFactory<>("vp"));
vp.setSortable(false);

leftTable.getColumns().setAll(num, name, vp);
leftTable.setItems(getLeftTableData());
}

private ObservableList getLeftTableData(){
ObservableList data = FXCollections.observableArrayList();
for(int i = 0; i < 5; i++){
    data.add(new Cell(i, 0));
}
return data;
}

private ObservableList getData(){
ObservableList data = FXCollections.observableArrayList();
for(int i = 0; i < 5; i++){
    data.add(new Cell(i, 0, 5));
}
return data;
}

@FXML
private void onEdit(MouseEvent event){
    cell_i = table.getFocusModel().getFocusedCell().getRow();
    cell_j = table.getFocusModel().getFocusedCell().getColumn();
    int value = (int) Main.getCriterion(0).getValue(cell_i, cell_j);
    if(value < 0) value = (int) Main.getCriterion(0).getValue(cell_j,
cell_i);
    slider.setValue(value);
    if(cell_i == cell_j){
        rb1.setDisable(true);
        rb2.setDisable(true);
        rb_all.setDisable(false);
        slider.setDisable(true);
        rb_all.setSelected(true);
        slider.setValue(1);
    }
    else{
        rb1.setDisable(false);
        rb2.setDisable(false);
        rb_all.setDisable(false);
        if(!rb_all.isSelected()) slider.setDisable(false);
    }
    rb1.setText(Main.getKr(cell_i));
    rb2.setText(Main.getKr(cell_j));
}

@FXML
private void rbClicked(MouseEvent event){
    if(rb1.isSelected()){
        reversed = false;
        slider.setDisable(false);
    }
    else if(rb2.isSelected()){
        reversed = true;
        slider.setDisable(false);
    }
    else if(rb_all.isSelected()){

```

```

        slider.setDisable(true);
        slider.setValue(1);
        sliderEvent(event);
    }
}

private void update() {
    TableColumn tc =
table.getFocusModel().getFocusedCell().getTableColumn();
    table.setItems(getData());
    table.getSelectionModel().select(cell_i, tc);
    leftTable.setItems(getLeftTableData());
    setLabels();
}

private void setLabels() {
    lambdaLabel.setText("λ = " +
(double) Math.round(Main.getCriterion(0).getHmax() * 1000) / 1000);
    ISLabel.setText("MC = " +
(double) Math.round(Main.getCriterion(0).getIS() * 1000) / 1000);
    OSLabel.setText("OC = " +
(double) Math.round(Main.getCriterion(0).getOS() * 1000) / 1000);
}

@FXML
private void sliderEvent(MouseEvent event) {
    int i = cell_i, j = cell_j;
    if(reversed) {
        i ^= j; j ^= i; i ^= j;
    }
    long value = Math.round(slider.getValue());
    slider.setValue(value);
    Main.getCriterion(0).change(i, j, value);
    update();
}

@FXML
private void initialize() {
    initTable();
    initLeftTable();
}

@FXML
private void onTimeClicked() {
    Main.changeScene(Parents.getTime());
}

@FXML private void onDiffClicked() {
    Main.changeScene(Parents.getDiff());
}

@FXML private void onCostClicked() {
    Main.changeScene(Parents.getCost());
}

@FXML private void onMinClicked() {
    Main.changeScene(Parents.getMin());
}

@FXML private void onQualityClicked() {
    Main.changeScene(Parents.getQuality());
}
}

```

```

@FXML private void onOkClicked(){
    Main.updateDiagram();
    Main.changeScene(Parents.getDiagram());
}

@FXML
private void onExit(MouseEvent event){
    System.exit(0);
}
}

```

Class Parents

```

import javafx.scene.Parent;

public class Parents {

    private static Parent aim;
    private static Parent time;
    private static Parent diff, cost, min, quality;
    private static Parent diagram;

    public static Parent getAim() {
        return aim;
    }

    public static void setAim(Parent aim) {
        Parents.aim = aim;
    }

    public static Parent getTime() {
        return time;
    }

    public static void setTime(Parent time) {
        Parents.time = time;
    }

    public static Parent getDiff() {
        return diff;
    }

    public static void setDiff(Parent diff) {
        Parents.diff = diff;
    }

    public static Parent getCost() {
        return cost;
    }

    public static void setCost(Parent cost) {
        Parents.cost = cost;
    }

    public static Parent getMin() {
        return min;
    }

    public static void setMin(Parent min) {
        Parents.min = min;
    }

    public static Parent getQuality() {

```

```

        return quality;
    }

    public static void setQuality(Parent quality) {
        Parents.quality = quality;
    }

    public static Parent getDiagram() {
        return diagram;
    }

    public static void setDiagram(Parent diagram) {
        Parents.diagram = diagram;
    }
}

```

Class DiagramController

```

import javafx.collections.FXCollections;
import javafx.collections.ObservableList;
import javafx.fxml.FXML;
import javafx.geometry.Side;
import javafx.scene.Parent;
import javafx.scene.chart.PieChart;
import javafx.scene.layout.GridPane;

import java.awt.*;

public class DiagramController {

    private double Q1, Q2, Q3;
    private LabeledPieChart pieChart;
    @FXML
    private GridPane gridPane;

    public void update() {
        Main.calcQ();
        Q1 = Main.getQ1();
        Q2 = Main.getQ2();
        Q3 = Main.getQ3();
        ObservableList<PieChart.Data> data = FXCollections.observableArrayList(
            new PieChart.Data("ПрофиШеф", (double)Math.round(Q1 * 1000) /
1000),
            new PieChart.Data("Русская чайная компания",
(double)Math.round(Q2 * 1000) / 1000),
            new PieChart.Data("Tea point", (double)Math.round(Q3 * 1000) /
1000)
        );
        pieChart.getData().clear();
        pieChart.getData().addAll(data);
        pieChart.setLegendSide(Side.RIGHT);
    }

    @FXML
    private void onOkClicked() {
        Main.changeScene(Parents.getAim());
    }

    @FXML
    private void initialize() {
        pieChart = new LabeledPieChart();
        gridPane.add(pieChart, 0, 0);
        update();
    }
}

```



```
}
```

Class LabeledPieChart

```
import java.util.HashMap;
import java.util.Map;
import javafx.collections.ListChangeListener;
import javafx.scene.chart.PieChart;
import javafx.scene.layout.Region;
import javafx.scene.shape.Arc;
import javafx.scene.text.Text;

public class LabeledPieChart extends PieChart {

    private final Map<Data, Text> _labels = new HashMap<>();

    public LabeledPieChart() {
        super();
        this.getData().addListener((ListChangeListener.Change<? extends Data> c)
-> {
            addLabels();
        });
    }

    @Override
    protected void layoutChartChildren(double top, double left, double
contentWidth, double contentHeight) {
        super.layoutChartChildren(top, left, contentWidth, contentHeight);

        double centerX = contentWidth / 2 + left;
        double centerY = contentHeight / 2 + top;

        layoutLabels(centerX, centerY);
    }

    private void addLabels() {

        for (Text label : _labels.values()) {
            this.getChartChildren().remove(label);
        }
        _labels.clear();
        for (Data vData : getData()) {
            final Text dataText;
            final double yValue = vData.getPieValue();
            dataText = new Text(Double.toString(yValue));
            _labels.put(vData, dataText);
            this.getChartChildren().add(dataText);
        }
    }

    private void layoutLabels(double centerX, double centerY) {

        double total = 0.0;
        for (Data d : this.getData()) {
            total += d.getPieValue();
        }
        double scale = (total != 0) ? 360 / total : 0;

        for (Map.Entry<Data, Text> entry : _labels.entrySet()) {
            Data vData = entry.getKey();
            Text vText = entry.getValue();
```

```

        Region vNode = (Region) vData.getNode();
        Arc arc = (Arc) vNode.getShape();

        double start = arc.getStartAngle();

        double size = (isClockwise()) ? (-scale *
Math.abs(vData.getPieValue())) : (scale * Math.abs(vData.getPieValue()));
        final double angle = normalizeAngle(start + (size / 2));
        final double sproutX = calcX(angle, arc.getRadiusX() / 2, centerX);
        final double sproutY = calcY(angle, arc.getRadiusY() / 2, centerY);

        vText.relocate(
            sproutX - vText.getBoundsInLocal().getWidth(),
            sproutY - vText.getBoundsInLocal().getHeight());
    }

}

private static double normalizeAngle(double angle) {
    double a = angle % 360;
    if (a <= -180) {
        a += 360;
    }
    if (a > 180) {
        a -= 360;
    }
    return a;
}

private static double calcX(double angle, double radius, double centerX) {
    return (double) (centerX + radius * Math.cos(Math.toRadians(-angle)));
}

private static double calcY(double angle, double radius, double centerY) {
    return (double) (centerY + radius * Math.sin(Math.toRadians(-angle)));
}
}

```