

**Международная Научно-Исследовательская Федерация
«Общественная наука»**

ТЕНДЕНЦИИ РАЗВИТИЯ НАУКИ И ОБРАЗОВАНИЯ

Сборник научных трудов

**по материалам
XXII международной научной конференции**

31 января 2017 г.

ЧАСТЬ 1

LJOURNAL.RU

Самара 2017

УДК 001.1
ББК 60

Т34

Тенденции развития науки и образования. Сборник научных трудов, по материалам XXIII международной научно-практической конференции 31 января 2017 г. Часть 1 Изд. НИЦ «Л-Журнал», 2016. - 48с.

SPLN 001-000001-0093-5B
DOI 10.18411/lj-31-01-2017-1
IDSP 000001:lj-31-01-2017-1

В сборнике научных трудов собраны материалы из различных областей научных знаний. В данном издании приведены все материалы, которые были присланы на XXII международную научно-практическую конференцию **Тенденции развития науки и образования**

Сборник предназначен для научных работников, преподавателей, аспирантов и студентов.

Все материалы, размещенные в сборнике, опубликованы в авторском варианте. Редакция не вносила коррективы в научные статьи. Ответственность за информацию, размещенную в материалах на всеобщее обозрение, несут их авторы.

Информация об опубликованных статьях будет передана в систему Российского индекса научного цитирования (РИНЦ)

Электронная версия сборника доступна на сайте научно-издательского центра «Л-Журнал». Сайт центра: ljournal.ru

УДК 001.1
ББК 60

SPLN 001-000001-0093-5B

<http://ljournal.ru>

Содержание

РАЗДЕЛ I. ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ	5
Буткай Д.В., Струков П. С., Суконщиков А.А., Бахтенко Е.А. Задача поиска информационных ресурсов с применением подхода к обработке документов на синтаксическом уровне	5
Буткай Д.В., Струков П. С., Суконщиков А.А., Бахтенко Е.А. Реализация системы семантического поиска на основе метаописаний	7
Буткай Д.В., Струков П. С., Суконщиков А.А., Бахтенко Е.А. Математический аппарат для описания и графического представления контекстно-свободных грамматик искусственных языков: синтаксис формул без переменных	10
Буткай Д.В., Струков П. С., Суконщиков А.А., Бахтенко Е.А. Математический аппарат для описания и графического представления контекстно-свободных грамматик искусственных языков: логическая семантика формул без переменных	13
Лифинцев М. А., Мандрыкина Н. Е., Кичигина А. К., Свиридова И. В. Об автоматизации программного сравнения методов СЛАУ	17
Отрах В.В., Суконщиков А.А., Наимов А.Н., Бахтенко Е.А. Актуальные легитимные источники данных для построения мультиагентной интеллектуальной системы поддержки принятия решений по оперативному рисковому ранжированию контрагентов	19
Отрах В.В., Суконщиков А.А., Наимов А.Н., Бахтенко Е.А. Вопросы информационной логистики при построении мультиагентной интеллектуальной системы поддержки принятия решений по оперативному рисковому ранжированию контрагентов	21
Отрах В.В., Суконщиков А.А., Бахтенко Е.А. Система кластерного анализа для выделения групп дебиторов в составе мультиагентной интеллектуальной системы поддержки принятия решений по оперативному рисковому ранжированию контрагентов	24
Отрах В.В., Суконщиков А.А., Кочкин Д.В., Бахтенко Е.А. Система нечеткого вывода в составе мультиагентной интеллектуальной системы поддержки принятия решений по оперативному рисковому ранжированию контрагентов.....	29
Свиридова И.В., Лебединская А.А., Внукова З.А., Кичигина А.К., Зайцева Т.В. Разработка информационной подсистемы логистики на примере ООО ТД «Малахов +»	32
Смирнов А.В., Андрианов И.А., Суконщиков А.А., Бахтенко Е.А. Математическая модель оптимизации доставки товаров автотранспортом на разветвленной сети дорог для решения задач и кластеризации.....	35

Смирнов А.В., Андрианов И.А., Суконщиков А.А., Бахтенко Е.А. Математическая модель оптимизации доставки товаров автотранспортом на разветвленной сети дорог для решения задачи маршрутизации.....	37
Смирнов А.В., Андрианов И.А., Суконщиков А.А., Бахтенко Е.А. Математическая модель логистического управления циклом заказа с использованием факторного анализа и механизмов прогнозирования.....	39
Смирнов А.В., Андрианов И.А., Суконщиков А.А., Бахтенко Е.А. Разработка модели адаптивного управления исполнением заказа на основе вероятностно-статистического подхода в логистической концепции «точно-в-срок»	41
Смирнов А.В., Андрианов И.А., Суконщиков А.А., Бахтенко Е.А. Разработка иерархической классификации оптимизационных моделей управления цепями поставок	45

РАЗДЕЛ I. ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Буткай Д.В., Струков П. С., Суконщиков А.А., Бахтенко Е.А.

Задача поиска информационных ресурсов с применением подхода к обработке документов на синтаксическом уровне

Вологодский государственный университет
(Россия, Вологда)

doi: 10.18411/lj-31-01-2017-1-01

idsp: 000001:lj-31-01-2017-1-01

Рассмотрим задачу поиска информационных ресурсов. Заданы множество документов $D = \{d_i \mid i = 1 \dots n\}$, где n – количество документов коллекции, и описание информационной потребности q .

Требуется найти подмножество R множества документов D , которое состоит из релевантных документов для запроса q – результирующее множество.

Кроме текстовых документов также может выполняться поиск мультимедийных ресурсов (изображений, аудиозаписей, видеороликов и т.п.). Однако для этого требуется создавать текстовые описания, которые включаются во множество документов D в качестве входных ресурсов.

Описание процесса поиска информации представлено на рис. 1.

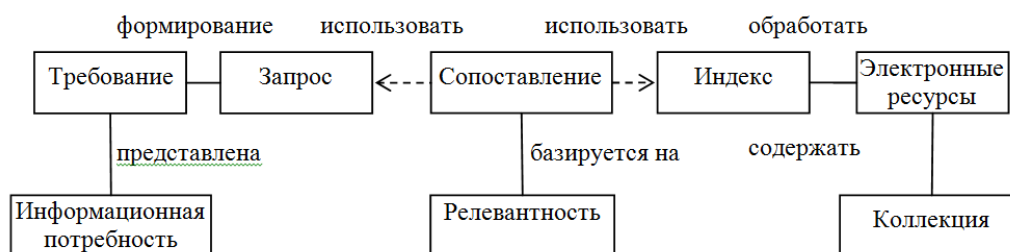


Рис. 1. - Процесс поиска информации

С одной стороны, у человека-пользователя есть информационная потребность, представляемая некоторым требованием, которое затем преобразуется в запрос. С другой стороны, в поисковых серверах имеются коллекции электронных ресурсов, которые индексируются для цели выполнения поиска. В результате обработки запросов поисковые серверы возвращают множество релевантных документов пользователям.

В настоящее время существует большое количество поисковых систем. Однако все они имеют общую архитектуру, которая представлена на рис. 2.



Рис. 2. - Общая архитектура систем информационного поиска

Как показано на рис. 2 любая система информационного поиска выполняет следующие три базовые функции:

1) Индексирование – сбор электронных ресурсов и создание их логических представлений, а также хранение этих представлений с использованием индексов (оптимизированных структур данных для быстрого выполнения поиска).

2) Формирование запросов – описания информационных потребностей пользователя на языке, поддерживаемом поисковой системой.

3) Сравнение – вычисления оценок близости (релевантности) между запросами и документами. На основе оценок релевантности определяется множество результатов, которое затем возвращается пользователям.

Связь между системными сообщениями и подсистемой формирования запроса означает возможность использования результатов поиска пользователями для уточнения информационных потребностей.

Система информационного поиска обычно реализуется на основе соответствующей теоретической модели, в которой описываются ее основные особенности: логические представления документов и запросов, а также алгоритмы вычисления оценки релевантности между ними. В общем случае модель информационного поиска состоит из следующих 4-х компонентов:

$$\text{Model} = [D, Q, F, R(q, d)], \quad (1)$$

где D – множество логических представлений документов (логические представления документов являются их упрощенными моделями в поисковых системах); Q – множество логических представлений информационных потребностей (запросов); F – платформа для моделирования документов, запросов и для реализации функции вычисления близости между документами и запросами – функции $R(d, q)$.

Подход к обработке документов на синтаксическом уровне в большинстве существующих моделей (например, модель булевого поиска, модель векторного пространства, вероятностная модель, модель нечеткого множества, модели обобщенного векторного пространства) вызывает следующие фундаментальные проблемы, которые затрудняют работу пользователей при поиске нужной информации: 1) синонимичность терминов; 2) многозначность терминов.

Кроме того, из-за отсутствия семантических отношений между терминами системы не способны найти документы, которые являются релевантными по смыслу, но не содержат термины запроса. В связи с этой проблемой снижается полнота множества результатов.

Решение этих проблем требует выполнять работу с семантикой документов и представлять документы на более высоком концептуальном уровне с использованием понятий из семантических моделей знаний предметных областей. В дополнение, по результату выполненного обзора и классификация существующих подходов семантического поиска с целью формирования общего представления о данном направлении информационного поиска нами отмечено, что важной особенностью подходов семантического поиска является использование концептуальных представлений документов, которые создаются на основе семантических моделей знаний предметных областей, а среди существующих инструментов представления знаний онтология считается наиболее выразительным. В онтологиях знания предметных областей описываются с помощью иерархий понятий и свойств, а также семантических сетей связанных экземпляров понятий.

Только использование онтологических моделей знаний позволит достигнуть значительного улучшения качества поиска по сравнению с существующими подходами. В связи с этим, несмотря на сложность и трудоемкость создания онтологических моделей, охватывающих области знаний поисковой системы, а также сложность и трудоемкость создания семантических описаний существующих текстов, считается, что семантический поиск на основе онтологий является наиболее перспективным направлением развития методов информационного поиска.

Буткай Д.В., Струков П. С., Суконщиков А.А., Бахтенко Е.А.
Реализация системы семантического поиска на основе метаописаний

*Вологодский государственный университет
 (Россия, Вологда)*

doi: 10.18411/lj-31-01-2017-1-02

idsp: 000001:lj-31-01-2017-1-02

Рассмотрим задачу реализации системы семантического поиска на основе метаописаний. Предложена общая схема системы семантического поиска, которая показана на рис. 1.



Рис. 1. - Общая схема системы семантического поиска с использованием метаописаний

Базовыми компонентами системы являются онтологии предметных областей. Такие онтологии могут быть созданы ручным путем с помощью редакторов онтологий, например Protégé, либо автоматическим путем.

В отличие от классических систем поиска информации, в предлагаемой схеме при индексировании и формировании запросов создаются метаописания на основе онтологий предметных областей, а обработка запросов выполняется на основе оценок семантической близости между метаописаниями.

Для повышения эффективности создания семантических метаописаний документов нами предложен полуавтоматический метод, согласно которому при создании метаописаний документов система автоматически выделяет элементы онтологии, а человек с помощью редактора выбирает необходимые данные для составления метаописаний. Предложенный метод автоматического выделения элементов онтологии

выполняется двумя этапами: поиск меток элементов онтологии в тексте; разрешение многозначности.

Проблема поиска меток элементов онтологии в тексте сформулирована следующим образом: заданы текстовый документ D и набор меток элементов онтологии Dic , состоящий из m текстовых меток:

$$Dic = \{метка_1, метка_2, \dots, метка_m\}.(1)$$

При этом документ D и каждая метка может быть представлена в виде последовательности лемм (лемма – это основная форма слова).

$D = лемма_1 \rightarrow лемма_2 \rightarrow \dots \rightarrow лемма_n$ и метка $i = лемма_1 \rightarrow \dots \rightarrow лемма_{k(i)}$, где n – количество лемм документа D ; $k(i)$ – количество лемм метки i .

Каждая j -я лемма документа D и j -я лемма i -й метки обозначаются соответственно как $D(j)$ и метка $i(j)$.

Требуется найти все пары индексов (i, j) , для которых строковое значение является элементом множества меток Dic и j является максимальным значением среди возможных для фиксированного индекса i .

$$строка(i, j, D) = D(i) \rightarrow D(i + 1) \rightarrow \dots \rightarrow D(j) \quad (2)$$

Блок-схема предложенного алгоритма решения поставленной задачи показана на рис. 2. Функция $search(D, i)$ либо возвращает максимальное значение индекса $j \geq i$, если существуют метки на i -й позиции документа D , либо -1 , если не существуют метки на i -й позиции документа D .

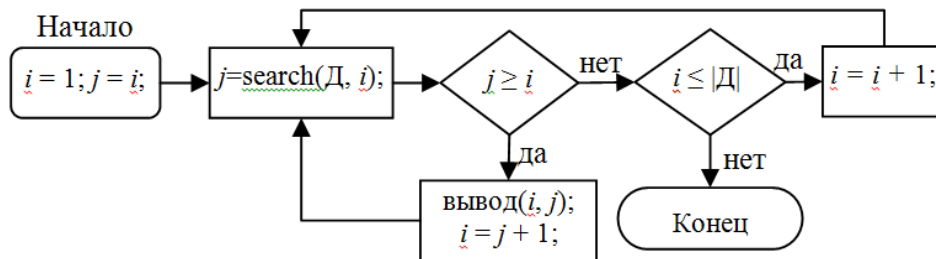


Рис. 2. - Блок-схема алгоритма поиска меток элементов

Для оптимизации выполнения функции $search(D, i)$ при проверке принадлежности метки набору меток предполагается использовать структуру данных, представленную на рис. 3.

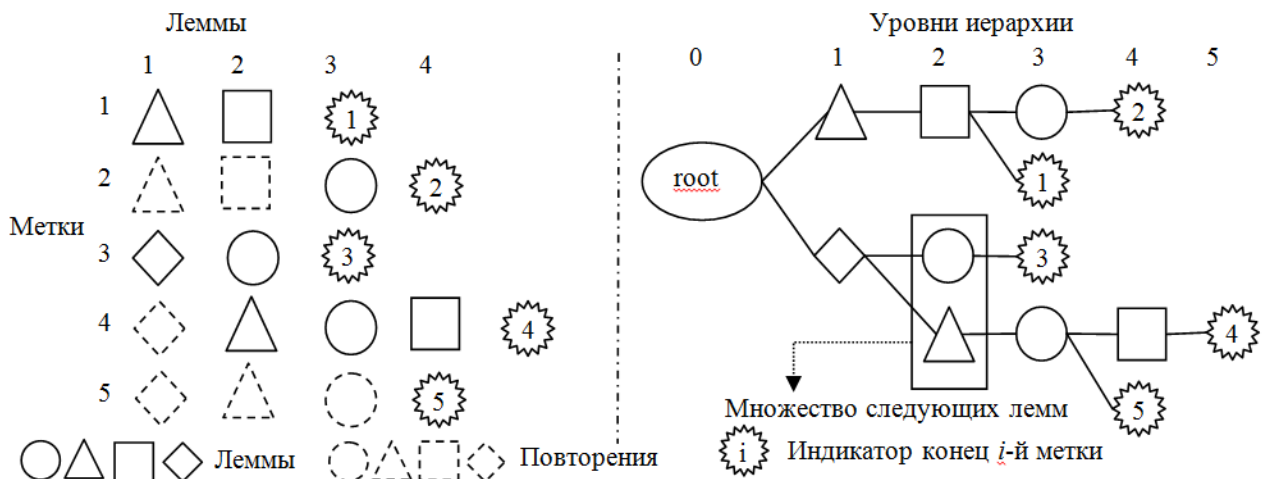


Рис. 3. - Структура данных наборов меток

На основе предложенной структуры данных имеется следующее пошаговое описание алгоритма функции $search(D, i)$:

Шаг 1. Начало алгоритма, $j = i$; $j_{max} = -1$; $pointer = root$;
 Шаг 2. $pointer = pointer.next(D(j))$ – переход на следующий узел, содержащий лемму $D(j)$.

Если существует узел p
 обновить значение $j_{max} = j$; $j = j + 1$; повторить шаг 2;
 иначе если $pointer \neq null$, то:
 $j = j + 1$; повторить шаг 2;
 иначе на шаг 3.

$\in pointer.next$

Шаг 3: вывод j_{max} ; конец алгоритма.

Следующим шагом после поиска меток является разрешение многозначности, т. е. определение подходящего элемента онтологии для каждой метки. Для этой цели нами были рассмотрены два существующих метода разрешения многозначности: разрешение многозначности на основе оценок семантической близости; разрешение многозначности на основе степени популярности.

Для обеспечения функционирования предложенной системы семантического поиска требуется создание индексов, показанных на рис. 4.

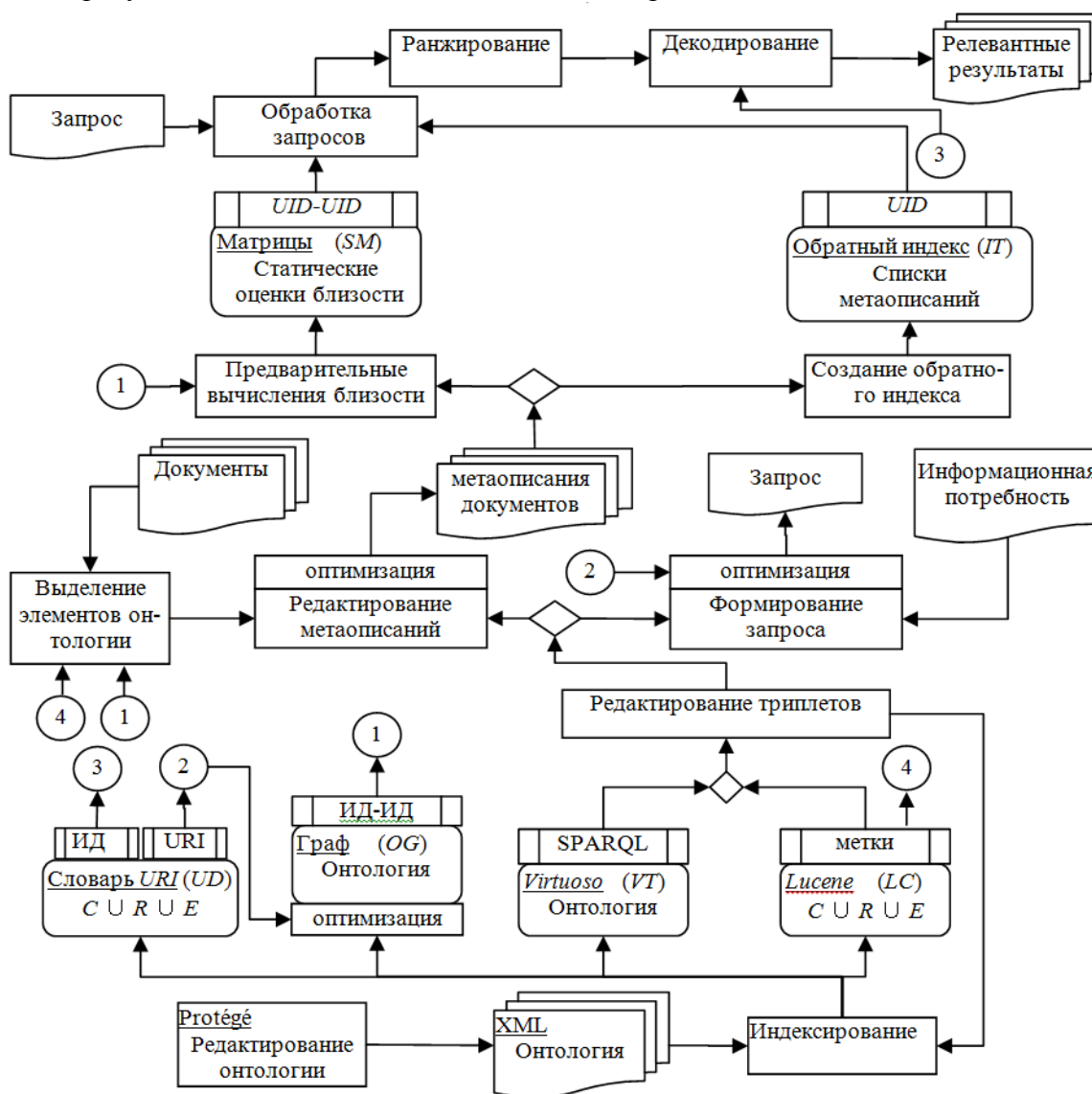


Рис. 4. - Структура индексов предложенной системы семантического поиска

Как показано на рис. 4, при индексировании онтологий создаются индексы UD, OG, VT и LC, а при индексировании метаописаний электронных ресурсов создаются индексы SM и IT. Индексы VT и LC используются в основном функциями формирования

триплетов для выборов элементов онтологии, а индекс OG используется для нахождения путей между элементами онтологии, на основе которых вычисляются их близости.

В индексе SM сохраняются оценки близости между элементами онтологии для цели исключения их повторных вычислений, а в индексе IT сохраняются инвертированные списки метаописаний для выполнения их фильтрации при обработке запросов.

На основе индекса UD разработан метод оптимизации хранения коллекции триплетов с использованием числовых идентификаторов, предназначенный для сокращения требуемых объемов памяти при хранении наборов триплетов. Основная идея предложенного метода заключается в замене каждого URI-идентификатора на уникальный числовой идентификатор, который затем может быть преобразован обратно в строку URI. Сопоставления между URI-идентификаторами и числовыми идентификаторами сохраняются в словаре URI-идентификаторов, структура которого показана на рис. 5.

Словарь URI-идентификаторов без сжатия		после сжатия			
i	строки URI в явном виде	d[i]	p[i]	ид	строки URI после сжатия
1	http://dbpedia.org/resource/Alexei_Gusarov	0	1	1	http://dbpedia.org/resource/ Alexei_Gusarov
2	http://dbpedia.org/resource/James_R._Russell	28	1	2	James_R._Russell
3	http://dbpedia.org/resource/Martina_Hellmann	28	1	3	Martina_Hellmann

Рис. 5. Структура словаря URI с использованием метода сжатия

С целью сокращения требуемого объема оперативной памяти для хранения словаря URI-идентификаторов разработан специальный метод сжатия без потери для набора строк URI и соответствующий метод декодирования. Как показано на рис. 5, с использованием разработанного метода сжатия потребуется меньше памяти для сохранения строки URI за счет удаления их общих фрагментов (например, если все строки URI начинаются с `http://...`, `data://...` и т.п.).

**Буткай Д.В., Струков П. С., Суконщиков А.А., Бахтенко Е.А.
Математический аппарат для описания и графического представления
контекстно-свободных грамматик искусственных языков:
синтаксис формул без переменных**

*Вологодский государственный университет
(Россия, Вологда)*

*doi: 10.18411/lj-31-01-2017-1-03
idsp: 000001:lj-31-01-2017-1-03*

Рассмотрим определение искусственного языка и его математической модели по аналогии с определением формального языка и его математической модели. Формальный язык определяется как множество текстов, которые могут быть написаны на этом языке. Искусственный язык определим как множество смыслов, которые могут быть выражены на этом языке.

Математической моделью текста является конечная строка, состоящая из знаков, принадлежащих заданному конечному набору. Таким образом, математической моделью (экстенсионалом) формального языка является некоторое (бесконечное) множество таких строк. Будем считать, как это принято в области искусственного интеллекта, математической моделью смысла (некоторой информации) конечную семантическую сеть, т.е. ориентированный размеченный связный граф без циклов, содержащий конечное число вершин. Способы представления смыслов (информации) семантическими сетями определяются правилами, специфическими для каждого искусственного языка; их рассмотрение выходит за рамки данной работы. Таким образом, математической моделью

(экстенционалом) искусственного языка будем считать некоторое (бесконечное) множество конечных семантических сетей.

Грамматикой, порождающей некоторый формальный язык, является исчисление, порождающее экстенционал этого формального языка. Грамматикой, порождающей некоторый искусственный язык, будем считать исчисление, порождающее экстенционал этого искусственного языка. Грамматика искусственного языка может быть дополнена набором правил (проекцией), позволяющим перейти от смысла (информации) к тексту (выражающему эту информацию), т.е. позволяющим по семантической сети построить соответствующую ей конечную строку из заданного конечного набора знаков.

Для представления грамматик искусственных языков требуется некоторый метаязык. Поскольку такие грамматики должны определять (бесконечные) подмножества множества семантических сетей, в качестве метаязыка естественно взять логический язык. В этом случае грамматика искусственного языка есть логическая формула этого метаязыка, истинная для тех и только тех семантических сетей, которые принадлежат этому искусственному языку. Поскольку логическая семантика метаязыка (способ вычисления значения истинности его формул) будет определена относительно семантических сетей, т.е. объектов, имеющих структуру, естественно ввести структуру и в синтаксис метаязыка – формулами метаязыка будут определенным образом размеченные графы (с циклами), в каждом из которых выделена одна вершина, называемая начальной вершиной этой формулы.

Наряду с логической семантикой метаязыка будет определена и его порождающая семантика – начальным состоянием порождающего процесса является начальная вершина (корень) порождаемой сети, которая объявляется активной; с каждой вершиной порождаемой сети связывается формула метаязыка; на очередном шаге порождающего процесса из множества активных вершин выбирается одна, из которой выполняется очередной шаг порождения в соответствии с формулой, связанной с этой вершиной; при этом вершина, из которой выполнен шаг порождения, перестает быть активной, но могут возникнуть новые активные вершины; порождающий процесс заканчивается, когда в порождаемой сети нет ни одной активной вершины. Логическая и порождающая семантики метаязыка связаны следующим естественным условием – по каждой формуле метаязыка может быть порождено множество тех и только тех семантических сетей, относительно которых эта формула истинна.

В свою очередь иерархическая семантическая сеть есть связный ориентированный граф без циклов, в котором каждая дуга имеет метку, а вершины могут быть одного из двух типов — простые и структурные; одна из вершин сети, в которую не входит ни одна дуга, выделяется и называется ее корнем. Каждая простая терминальная вершина сети имеет в качестве метки константу некоторого сорта. Корень сети имеет две метки-термина, из которых первая метка есть метка искусственного языка (класса), а вторая – индивидуальная метка сети. Каждая структурная вершина сети является контейнером, содержащим упорядоченное конечное множество иерархических семантических сетей (далее по тексту — «сеть») с одной и той же меткой класса и попарно различными индивидуальными метками.

Контекстно-свободные грамматики искусственных языков. Контекстно-свободная грамматика искусственного языка есть формула без переменных метаязыка, начальная вершина которой имеет метку – название этого искусственного языка.

Синтаксис формул без переменных. Формула без переменных метаязыка может относиться к одной из следующих групп:

- простая формула без переменных;
- простая кванторная формула без переменных;
- унарная формула без переменных;
- пропозициональная формула без переменных;

- структурная кванторная формула без переменных.

Простая формула без переменных есть граф, состоящий из единственной вершины с меткой c (рис. 1). Эту единственную вершину будем называть начальной вершиной простой формулы без переменных. Метка является константой.



Рис. 1. - Простая формула без переменных

Простая кванторная формула без переменных есть граф, состоящий из единственной вершины с меткой, имеющей вид QMT , где Q - знак квантора, M - описание (конечного или бесконечного) множества, а T - термин (рис. 2). Знаком квантора может быть: \forall (для всех), \exists (существует), $\exists 2$ (существует не менее двух), $\exists ?$ (существует, но не для всех), $\exists !$ (существует и единственен), $\exists []$ (существует подынтервал). Множество кванторов является расширяемым.



Рис. 2. - Простая кванторная формула без переменных

Описание конечного множества может иметь вид $\{c_1, \dots, c_n\}$, где c_1, \dots, c_n - попарно различные константы, либо быть целым конечным интервалом; в этом случае знаком квантора может быть $\forall, \exists, \exists 2, \exists ?$ или $\exists !$. Описание бесконечного множества может быть названием сорта, неименованным множеством «*», или вещественным конечным интервалом; в этом случае знаком квантора может быть $\exists, \exists 2, \exists !$. Если знак квантора есть $\exists []$, то описанием множества может быть только конечный целый или вещественный интервал.

Названием сорта может быть: "строка", "целый", "вещественный", "целый интервал", "вещественный интервал", "дата-время". Множество сортов является расширяемым.

Унарная формула без переменных есть граф, состоящий из начальной вершины и дуги с меткой T (термином), выходящей из начальной вершины и входящей в начальную вершину некоторой формулы без переменных F (рис. 3).

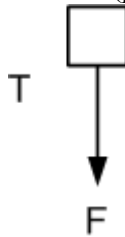


Рис. 3. - Унарная формула без переменных

Пропозициональная формула без переменных есть граф, состоящий из начальной вершины с пропозициональной меткой P и выходящих из нее n дуг (не менее двух), каждая из которых имеет метку T_i (термин; i от 1 до n ; все эти метки должны быть попарно различны) и входит в начальную вершину некоторой формулы без переменных F_i (рис. 4).

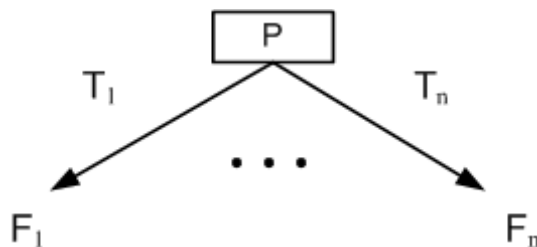


Рис. 4. - Пропозициональная формула без переменных

Пропозициональными метками Р являются:

& - конъюнкция;

∨ - дизъюнкция;

| - исключающее или.

Множество пропозициональных меток является расширяемым.

Любая из дуг, выходящая из начальной вершины с пропозициональной меткой &, может иметь метку факультативности "[]".

Структурная кванторная формула без переменных состоит из начальной вершины с меткой, имеющей вид QMT, где Q — знак квантора, M — описание (конечного или бесконечного) множества, а T — термин, и формулы без переменных F, начальная вершина которой изображается внутри начальной вершины структурной кванторной формулы без переменных (рис. 5).

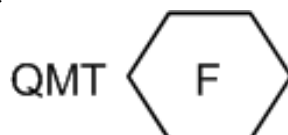


Рис. 5. - Структурная кванторная формула без переменных

Описание конечного множества может иметь вид $\{c_1, \dots, c_n\}$, где c_1, \dots, c_n — попарно различные константы, либо быть целым конечным интервалом; в этом случае знаком квантора может быть $\forall, \exists, \exists 2, \exists ?$. Описание бесконечного множества может быть названием сорта, неименованным множеством «*», или вещественным конечным интервалом; в этом случае знаком квантора может быть $\exists, \exists 2$.

**Буткай Д.В., Струков П. С., Суконщиков А.А., Бахтенко Е.А.
Математический аппарат для описания и графического представления
контекстно-свободных грамматик искусственных языков:
логическая семантика формул без переменных**

*Вологодский государственный университет
(Россия, Вологда)*

doi: 10.18411/lj-31-01-2017-1-04

idsp: 000001:lj-31-01-2017-1-04

Рассмотрим логическую семантику формул без переменных. Формула без переменных F метаязыка истинна относительно сети N, если F истинна на корне R сети N, а метка входной вершины формулы без переменных F совпадает с меткой класса корня R сети N.

Простая формула без переменных F (рис. 1) истинна на вершине V сети N, если V является терминальной и простой, а ее меткой является константа с.



Рис. 1. - Семантика простой формулы без переменных

Если в простой кванторной формуле без переменных описание конечного множества имеет вид $\{c_1, \dots, c_n\}$, где c_1, \dots, c_n — попарно различные константы, то конечное множество состоит из элементов c_1, \dots, c_n ; если же описание конечного множества есть конечный целый интервал, то конечное множество состоит из всех целых констант, принадлежащих этому интервалу. Если описание бесконечного множества есть название сорта, то бесконечное множество состоит из констант, представляющих все значения этого сорта; если описание бесконечного множества есть «*», то бесконечное множество состоит из всех констант специального вида. Если описание бесконечного

множества есть конечный вещественный интервал, то бесконечное множество состоит из всех вещественных констант, принадлежащих этому интервалу.

Если метка входной вершины простой кванторной формулы без переменных F (рис. 2) имеет вид $\forall FS T$, где FS – описание конечного множества $\{d_1, \dots, d_n\}$, а T – термин, то F истинна на вершине V сети N , если V является структурной и терминальной вершиной, содержащей n сетей, каждая из которых состоит из одного корня с меткой класса - T и индивидуальными метками – элементами множества $\{d_1, \dots, d_n\}$.



Рис. 2. - Семантика простой кванторной формулы без переменных с квантором \forall \square

Если метка входной вершины простой кванторной формулы без переменных F (рис. 3) имеет вид $\exists M T$, где M – описание (конечного или бесконечного) множества, а T – термин, то F истинна на вершине V сети N , если V является структурной и терминальной вершиной, содержащей непустое множество сетей, каждая из которых состоит из одного корня с меткой класса - T и индивидуальной меткой – одним из элементов множества (индивидуальные метки попарно различны).



Рис. 3. - Семантика простой кванторной формулы без переменных с квантором \exists \square

Если метка входной вершины простой кванторной формулы без переменных F (рис. 4) имеет вид $\exists 2 M T$, где M – описание (конечного или бесконечного) множества, а T – термин, то F истинна на вершине V сети N , если V является структурной и терминальной вершиной, содержащей не менее двух сетей, каждая из которых состоит из одного корня с меткой класса - T и индивидуальной меткой – одним из элементов множества (индивидуальные метки попарно различны).



Рис. 4. - Семантика простой кванторной формулы без переменных с квантором $\exists 2$ $\square\square$

Если метка входной вершины простой кванторной формулы без переменных F (рис. 5) имеет вид $\exists ? FS T$, где FS – описание конечного множества $\{d_1, \dots, d_n\}$, а T – термин, то F истинна на вершине V сети N , если V является структурной и терминальной вершиной, содержащей менее, чем n сетей, каждая из которых состоит из одного корня с меткой класса - T и попарно различными индивидуальными метками - элементами множества $\{d_1, \dots, d_n\}$.

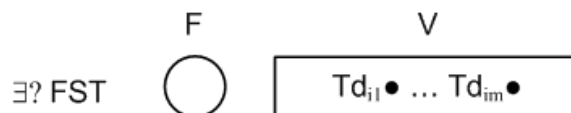


Рис. 5. - Семантика простой кванторной формулы без переменных с квантором $\exists ?$ $\square\square$

Если метка входной вершины простой кванторной формулы без переменных F (рис. 6) имеет вид $\exists! M T$, где M – описание (конечного или бесконечного) множества, а T – термин, то F истинна на вершине V сети N , если V является простой и терминальной вершиной, а ее меткой является один из элементов множества.



Рис. 6. - Семантика простой кванторной формулы без переменных с квантором □□□

Если метка входной вершины простой кванторной формулы без переменных F (рис. 7) имеет вид $\exists [] I T$, где I – конечный целый (вещественный) интервал, а T – термин, то F истинна на вершине V сети N , если V является простой и терминальной вершиной, а ее меткой является целый (вещественный) интервал, являющийся подынтервалом интервала I .

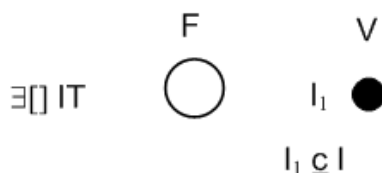


Рис. 7. - Семантика простой кванторной формулы без переменных с квантором □□□

Унарная формула без переменных F (рис. 8) истинна на вершине V сети N , если V является простой вершиной, из нее выходит в точности одна дуга с меткой T и эта дуга входит в вершину V_1 , на которой истинна формула без переменных F_1 .

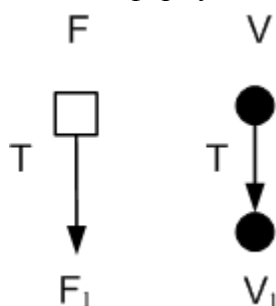


Рис. 8. - Семантика унарной формулы без переменных

Если пропозициональной меткой начальной вершины W пропозициональной формулы без переменных F (рис. 9) является $\&$, а некоторые дуги, выходящие из W , имеют метки $[\]$, то F истинна на вершине V сети N , если V является простой вершиной, для каждой дуги с меткой T_i , выходящей из V , существует дуга с такой же меткой T_i , выходящая из W , для каждой дуги с меткой T_i , выходящей из W и не имеющей метки $[\]$, существует дуга с такой же меткой T_i , выходящая из V , а каждая формула без переменных F_i истинна на вершине, в которую входит дуга с меткой T_i , выходящая из вершины V .

Если пропозициональной меткой начальной вершины W пропозициональной формулы без переменных F (рис. 10) является \vee , то F истинна на вершине V сети N , если V является простой вершиной, существует подмножество дуг, выходящих из W , между которым и множеством дуг, выходящих из V , существует взаимно-однозначное соответствие, при котором метки T_i соответствующих дуг совпадают, а формулы без переменных F_i , в начальные вершины которых входят дуги, выходящие из W , истинны на вершинах, в которые входят соответствующие дуги с метками T_i , выходящие из V .

Если пропозициональной меткой начальной вершины W пропозициональной формулы без переменных F (рис. 11) является $|$, то F истинна на вершине V сети N , если V является простой вершиной, из нее выходит единственная дуга, существует дуга,

выходящая из W , такая что метки T_i этих дуг совпадают, а формула без переменных F_i , в начальную вершину которой входит дуга с меткой T_i , выходящая из W , истинна на вершине, в которую входит дуга с меткой T_i , выходящая из V .

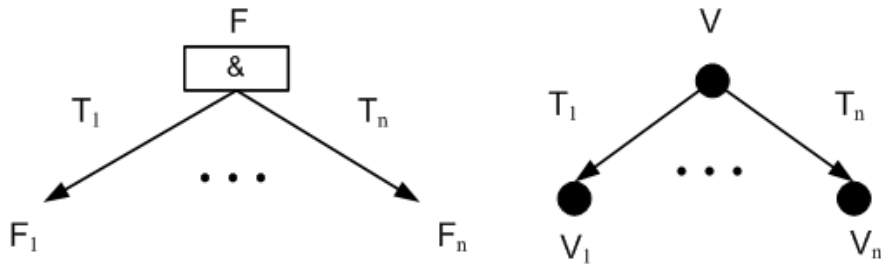


Рис. 9. - Семантика конъюнкции без переменных

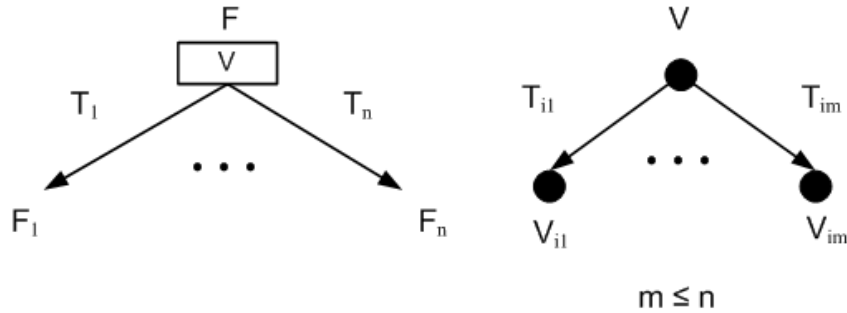


Рис. 10. - Семантика дизъюнкции без переменных

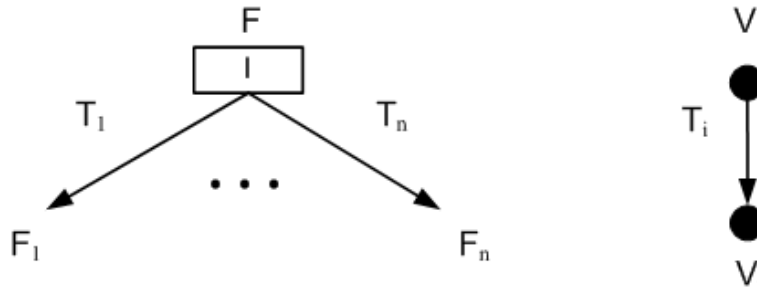


Рис. 11. - Семантика исключаящего ИЛИ без переменных

Если метка входной вершины структурной кванторной формулы без переменных F (рис. 12) имеет вид $\forall FS T$, где FS – описание конечного множества $\{d_1, \dots, d_n\}$, а T – термин, то F истинна на вершине V сети N , если V является структурной вершиной, содержащей n сетей, каждая из которых имеет метку класса - T и индивидуальные метки – попарно различные элементы множества $\{d_1, \dots, d_n\}$, и на корне каждой из которых истинна формула без переменных F_1 .

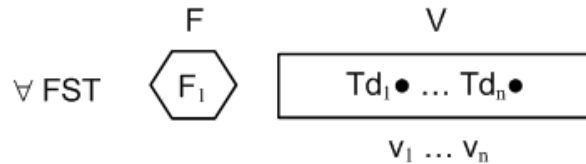


Рис. 12. - Семантика структурной кванторной формулы без переменных F с квантором \square

Если метка входной вершины структурной кванторной формулы без переменных F (рис. 13) имеет вид $\exists M T$, где M – описание (конечного или бесконечного) множества, а T – термин, то F истинна на вершине V сети N , если V является структурной вершиной, содержащей непустое множество сетей, каждая из которых имеет метку класса - T и индивидуальные метки – попарно различные элементы множества, и на корне каждой из которых истинна формула без переменных F_1 .

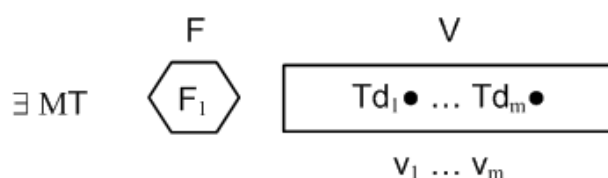


Рис. 13. - Семантика структурной кванторной формулы без переменных F с квантором \square

Если метка входной вершины структурной кванторной формулы без переменных F (рис. 14) имеет вид $\exists 2 M T$, где M – описание (конечного или бесконечного) множества, а T – термин, то F истинна на вершине V сети N , если V является структурной вершиной, содержащей не менее двух сетей, каждая из которых имеет метку класса - T и индивидуальные метки – попарно различные элементы множества, и на корне каждой из которых истинна формула без переменных $F1$.

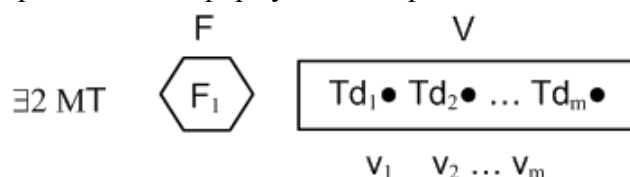


Рис. 14. - Семантика структурной кванторной формулы без переменных F с квантором $\square\square$

Если метка входной вершины структурной кванторной формулы без переменных F (рис. 15) имеет вид $\exists ? FS T$, где FS – описание конечного множества $\{d1, \dots, dn\}$, а T – термин, то F истинна на вершине V сети N , если V является структурной вершиной, содержащей менее, чем n сетей, каждая из которых имеет метку класса - T и индивидуальные метки – попарно различные элементы множества $\{d1, \dots, dn\}$, и на корне каждой из которых истинна формула без переменных $F1$.

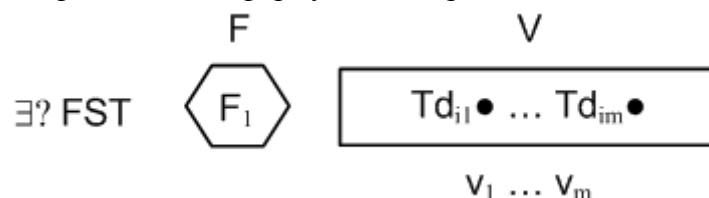


Рис. 15. - Семантика структурной кванторной формулы без переменных F с квантором $\square\square\square$

Лифинцев М. А., Мандрыкина Н. Е., Кичигина А. К., Свиридова И. В.
Об автоматизации программного сравнения методов СЛАУ

Белгородский государственный национальный исследовательский университет
(Россия, Белгород)

doi: 10.18411/lj-31-01-2017-1-05

idsp: 000001:lj-31-01-2017-1-05

Решение систем линейных алгебраических уравнений (СЛАУ) – одна из основных задач вычислительной линейной алгебры. Значительная часть численных методов решения различных задач включает в себя решение систем линейных уравнений как элементарный шаг соответствующего алгоритма.

В практических задачах часто бывает нужно найти решение, удовлетворяющее большому числу возможно противоречивых требований. Такая задача сводится к системе линейных уравнений

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2 \\ \dots \\ a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n = b_m \end{cases} \iff AX = B$$

при числе уравнений m большем числа неизвестных n , то такая переопределенная система, как правило, несовместна. В этом случае задача может быть решена только

путем выбора некоторого компромисса – все требования могут быть удовлетворены не полностью, а лишь до некоторой степени [2]. Псевдорешение системы $AX=B$ называется

столбец $X \in \mathbb{R}^n$, обеспечивающий минимум величины

$$\sum_{i=1}^m [a_{i1}x_1 + a_{i2}x_2 + \dots + a_{in}x_n - b_i]^2$$

Такому определению можно также соотнести вероятностную интерпретацию. Пусть для определения неизвестных величин $x_1 \dots x_n$ проводятся m экспериментов, описываемых линейными уравнениями:

$$a_{i1}x_1 + a_{i2}x_2 + \dots + a_{in}x_n = b_i \quad \text{при } i \in \{1, \dots, m\}$$

При этом величины $\{a_{ij}\}, i \in \{1, \dots, m\}, j \in \{1, \dots, n\}$ – известные постоянные, не подверженные сопутствующим экспериментам погрешностям, а вот величины $\{b_i\}_{i=1}^m$ этим погрешностям подвержены. Формально каждое из равенств следует рассматривать как приближенное. Понятно, что при таких обстоятельствах не имеет смысла гоняться за точным решением системы $AX=B$. Искать следует приближенное решение, оптимальное в некотором смысле.

Существует псевдорешение системы $AX=B$ и оно является решением системы $[A^T A]X=A^T B$. Это решение будет единственным тогда и только тогда, когда $\text{rank}A=n$.

Система $[A^T A]X=A^T B$ называется нормальной системой по отношению к системе $AX=B$. Формально она получается домножением системы $AX=B$ слева на матрицу A^T . Заметим также, что если $m=n$ и $\det A \neq 0$, то вседорешение системы совпадает с решением в традиционном смысле.

Если нормальная система имеет бесконечное количество решений, то обычно в качестве псевдорешения берут какое-то одно из них – как правило, у которого минимальна сумма квадратов компонент («длина»).

В качестве языка программирования для автоматизации программного сравнения СЛАУ был выбран язык C++. Это очень мощный язык, позволяющий создавать программы любого назначения и любой сложности[1].

Определившись с выбором подходящего компилятора, можно приступить к написанию самой программы. В начале работы с программой высвечивается форма, которая требует ввода размерности матрицы, а затем введения элементов.

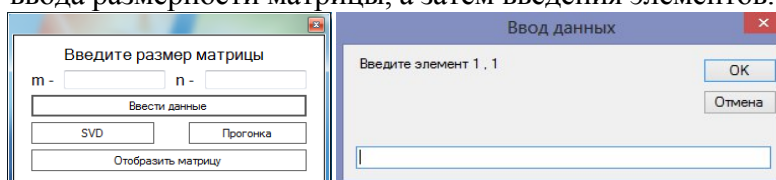


Рисунок 3, 4 – Формы требующие ввести размерность матрицы и ее элементы

После ввода всех необходимых данных в программу, проводится решение систем методом SVD и методом прогонки, которые выбираются пользователем самостоятельно.

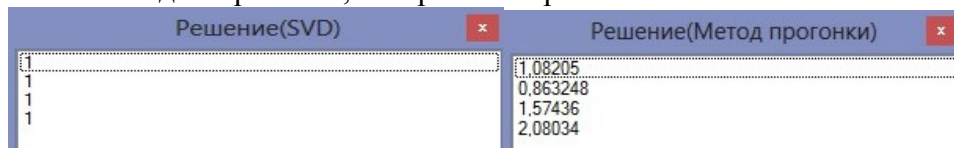


Рисунок 5, 6 – Решение методом SVD и методом прогонки

Время расчета метода прогонки – 1,7секунд. Время расчета метода SVD – 1,5 секунд.

Метод наименьших квадратов (SVD) дает точное решение системы алгебраических уравнений, но требует перед своим применением дополнительные преобразования – проверку на то, что количество строк больше количества столбцов.

Во-первых, количество строк сильно зависит от матрицы A исходной системы уравнений вида $Ax=b$. Чем больше количество строк в уравнении, тем больше времени требуется программе, чтобы найти корни данной системы. Во-вторых, на количество шагов влияет начальное приближение. Чем оно ближе к точному решению, тем меньше требуется шагов для сходимости метода.

В итоге можно сделать вывод: метод наименьших квадратов (SVD) является лучшим при решении системы линейных уравнений, он является менее объемным и требует меньше времени для нахождения корней, чем метод прогонки.

Список используемых источников информации

1. М. Шлее. Qt 4.8. Профессиональное программирование на C++. – СПб.: БХВ-Петербург, 2012. – с. 452
2. Численное решение переопределенных СЛАУ. Метод наименьших квадратов. [Электронный ресурс] – Режим доступа <http://www.intuit.ru/studies/courses/1012/168/lecture/4594>

¹Отрах В.В., ²Суконщиков А.А., ²Наимов А.Н., ²Бахтенко Е.А.

Актуальные легитимные источники данных для построения мультиагентной интеллектуальной системы поддержки принятия решений по оперативному рисковому ранжированию контрагентов

¹*Московский технологический институт
(Россия, Москва)*

²*Вологодский государственный университет
(Россия, Вологда)*

doi: 10.18411/lj-31-01-2017-1-06

idsp: 000001:lj-31-01-2017-1-06

Сталкиваясь с необходимостью проверить контрагента, сотрудники служб безопасности все чаще обращаются за помощью к специализированным продуктам и системам обработки информации. Базы данных по контрагентам можно разделить на легальные и нелегальные, по юридическим и физическим лицам. Нелегальные базы данных имеют несколько недостатков:

- Они нелегальны;
- Данные в таких базах, как правило, обладают слабой актуальностью;
- Охватывают такие базы данных не каждый регион.

Однако всю или почти всю необходимую информацию по юридическим лицам (контрагентам) можно найти в легальных базах данных, более того – в открытых источниках. Ведь эта информация не относится к разряду персональных данных и является публичной. Минус здесь только один – информацию нужно собирать из разных мест.

Для того, чтобы составить консолидированный портрет контрагента, рационально инициировать разработку корпоративной системы рискового ранжирования контрагентов на основе открытых источников. Разработка автоматизированной системы позволит избежать временных затрат и позволит оперативно обрабатывать большой набор данных в контексте деятельности организации.

Основные официальные источники:

- Федеральная налоговая служба (ЕГРЮЛ\ЕГРИП);
- Федеральная служба государственной статистики;
- Федеральный Арбитражный суд РФ;
- Федеральная служба судебных приставов;
- Генеральная прокуратура РФ;
- Центральный Банк РФ;

- Единый федеральный реестр сведений о фактах деятельности юридических лиц;
- Портал государственных закупок РФ;
- Электронные торговые площадки.

Основные группы данных:

- Регистрационные данные (Регистрационная информация ООО и ИП);
- Отчетность (Годовая и квартальная отчетность юридических лиц);
- Сведения о деятельности (Участие в тендерах, лицензии, проверки, набор персонала);
- Рисковые факторы (Банкротства, суды, смена владельцев).
- Типовые входные данные для построения отчета и ранжирования:
- Сведения о регистрации ФНС (Федеральная налоговая служба);
- Информация из вестника ЕГРЮЛ;
- Финансовую информацию – баланс, отчет о прибылях и убытках (Росстат)
- Коэффициенты доверия компании;
- Индекс финансового риска;
- Арбитражные дела компании (если есть);
- Сообщения о банкротствах (если были);
- Информация о руководителе;
- Совпадения и вероятные совпадения;
- Информация об органах управления предприятия;
- Данные о лицах, имеющих право действовать без доверенности;
- Структура компании;
- Данные о совладельцах;
- Данные о деятельности компании;
- Сведения о государственных заказах;
- Сведения о коммерческих заказах;
- Список лицензий;
- Информация об отраслевой принадлежности предприятия (ОКВЭД).

Рассмотрим правовую сторону использования и публикации данных в информационном пространстве компании. Используемая информация для рискованного ранжирования контрагента является открытой и общедоступной в соответствии со следующими нормативно-правовыми актами:

- Сведения из ЕГРЮЛ и ЕГРИП — п.1 ст.6 Федерального закона от 08.08.2001 № 129-ФЗ «О государственной регистрации юридических лиц и индивидуальных предпринимателей»;
- Бухгалтерская отчетность — п.42 Положения по бухгалтерскому учету «Бухгалтерская отчетность организации» (ПБУ 4/99), утвержденному Приказом Минфина РФ от 06.07.1999 N 43н;
- Информации о закупках для государственных и муниципальных нужд — ст.7 Федерального закона от 05.04.2013 № 44-ФЗ «О контрактной системе в сфере закупок товаров, работ, услуг для обеспечения государственных и муниципальных нужд»;
- Тексты судебных актов — ст.15 Федерального закона от 22.12.2008 № 262-ФЗ «Об обеспечении доступа к информации о деятельности судов в Российской Федерации»;
- Сведения из банка данных исполнительных производств — ст. 6.1 Федерального закона от 02.10.2007 №229-ФЗ «Об исполнительном производстве»;

- Ежегодный план проведения плановых проверок — ст. 9 Федерального закона от 26.12.2008 № 294-ФЗ «О защите прав юридических лиц и индивидуальных предпринимателей при осуществлении государственного контроля (надзора) и муниципального контроля»;
- Финансовая отчетность кредитных организаций — Указание ЦБ РФ от 03.12.2012 № 2923-У «О раскрытии и представлении головными кредитными организациями банковских групп консолидированной финансовой отчетности»;
- Сведения из ЕФРСДЮЛ — ст. 7.1 Федерального закона от 08.08.2001 № 129-ФЗ «О государственной регистрации юридических лиц и индивидуальных предпринимателей».

Согласно Федеральному закону от 27.07.2006 № 149-ФЗ «Об информации, информационных технологиях и защите информации»:

- Общедоступная информация может использоваться любыми лицами по их усмотрению при соблюдении установленных федеральными законами ограничений в отношении распространения такой информации (п. 2 ст.7);
- Информация может свободно использоваться любым лицом и передаваться одним лицом другому лицу, если федеральными законами не установлены ограничения доступа к информации, либо иные требования к порядку ее предоставления или распространения (п.1 ст.5).

¹Отрах В.В., ²Суконщиков А.А., ²Наимов А.Н., ²Бахтенко Е.А.

Вопросы информационной логистики при построении мультиагентной интеллектуальной системы поддержки принятия решений по оперативному рисковому ранжированию контрагентов

¹*Московский технологический институт
(Россия, Москва)*

²*Вологодский государственный университет
(Россия, Вологда)*

doi: 10.18411/lj-31-01-2017-1-07

idsp: 000001:lj-31-01-2017-1-07

Проверка деловых партнеров – мероприятие непростое, занимающее много времени, но гарантированно защищающее от излишних финансовых потерь. Облегчить задачу может построение корпоративной автоматизированной информационной системы рискованного ранжирования контрагентов, аккумулирующих сведения из открытых источников, которая позволяла бы осуществить комплексную проверку партнеров по бизнесу, в том числе выяснить, если контрагент не платит НДС. Таким образом, можно было бы оперативно выяснить задолженности контрагента, факты по привлечению его к ответственности, сведения об арбитражной практике, нет ли у компании массового учредителя и пр. Не стоит забывать, что построение такой системы не должно ограничиваться только данными из внешних источников и сводиться только к получению статистической сводки – необходимо использовать информацию, накопленную в самой организации. Только так можно производить эффективное ранжирование в зависимости от области деятельности организации ее и контрагентов, с учетом индивидуальных особенностей бизнеса, а также условий сотрудничества (договоры, доп. соглашения и т.д.). Поэтому особое внимание нужно уделить организации внутрикорпоративного информационного хранилища и базы знаний.

В итоге при рисковом ранжировании контрагента необходимо работать с данными, как из внешнего, так и из внутреннего по отношению к организации информационного пространства. А когда речь заходит о среднем или крупном бизнесе с разветвленной сетью филиалов, дочерних организаций и партнеров, в первую очередь необходимо решить вопросы информационной логистики в виду колоссального объема информационных единиц, подлежащих обработке.

Информационная логистика - часть логистики, которая организует поток информации, сопровождающий материальный поток в процессе его перемещения. Информационная логистика является связующим звеном между снабжением, производством и сбытом на предприятии [1]. Информационная логистика управляет процессами движения и складирования товаров на предприятии, обеспечивая своевременную доставку этих товаров в необходимом количестве, требуемой комплектации и нужного качества из места их производства до места потребления с минимальными затратами и оптимальным сервисом.

Информационный поток - это совокупность циркулирующих в логистической системе между логистической системой и внешней средой сообщений, необходимых для управления и контроля логистических операций. Информационный поток соответствует материальному и может существовать в виде бумажных и электронных документов.

Информационный поток может опережать материальный, следовать одновременно с ним или после него. При этом информационный поток может быть направлен как в одну сторону с материальным, так и в противоположную.

Опережающий информационный поток во встречном направлении содержит, как правило, сведения о заказе.

Опережающий информационный поток в прямом направлении - это предварительные сообщения о предстоящем прибытии груза.

Одновременно с материальным потоком идет информация в прямом направлении о количественных и качественных параметрах материального потока.

Вслед за материальным потоком во встречном направлении может проходить информация о результатах приемки груза по количеству или по качеству, разнообразные претензии, подтверждения.

Выделяют виды информационных потоков в зависимости от вида связываемых потоком систем:

- горизонтальный - поток сообщений между партнерами по хозяйственным связям одного уровня управления;
- вертикальный - поток сообщений, поступающих от руководства в подведомственные ему звенья логистической системы.

От места происхождения:

- внешний - поток, протекающий во внешней, по отношению к логистической системе, среде;
- внутренний - поток сообщений, циркулирующих внутри одной логистической системы либо одной из подсистем.
- От направления потока:
 - входной - поток сообщений, входящих в логистическую систему, либо в одну из подсистем;
 - выходной - поток сообщений, выходящих за пределы логистической системы либо одной из подсистем.

От срочности:

- обычные; срочные; очень срочные.

От степени секретности:

- обычные; содержащие коммерческую тайну; содержащие государственную тайну.

От значимости почтовых сообщений:

- простые; заказные; ценные.

От скорости передачи:

- традиционные (почта); быстрые (факс, эл.почта, телефон).

От области охвата:

- местные; иногородние; дальние; международные.

Информационные потоки в логистике формируются в соответствии с материальными. Но иногда материальный поток может прибыть в заданное место, а документы на него могут быть еще не доставлены. Такой материальный поток считается неотфактурованной поставкой, и принимается получателем на хранение до прибытия документов. Может быть и наоборот: документы прибывают на место назначения раньше самого груза.

Опережение информационным потоком материального предпочтительнее. Это дает возможность лучше подготовиться к приему грузов. Информационные потоки должны быть адекватны материальным потокам в части характеристики этих потоков.

Основная задача информационной логистики заключается в доставке информации к системе управления предприятием и от нее. Информационная логистика выполняет функции:

- собирает возникающую информацию;
- анализирует информацию;
- перемещает информацию;
- накапливает и хранит информацию;
- фильтрует поток информации (отбирает необходимые для того или иного уровня управления данные и документы);
- объединяет и разъединяет информационные потоки;
- выполняет элементарные информационные преобразования;
- управляет информационным потоком.

Успешная организация информационной логистики востребует автоматизированные соответствующие информационные системы. В их задачи входит:

- Постоянное обеспечение управляющих органов логистической системы достоверной информации о движении заказа.
- Постоянное обеспечение сотрудников предприятия адекватной информацией о движении продукции в режиме реального времени.
- Оперативное управление предприятием.
- Предоставление руководству наглядной информации об использовании инвестиций.
- Предоставление информации о расходах.
- Помощь в выявлении узких мест.
- Обеспечение возможности оценки сроков исполнения полученных заказов.
- Обеспечение прибыльной работы предприятия за счет оптимизации логистических процессов.

Список используемых источников информации

1. Алесинская Т.В. Основы логистики. Функциональные области логистического управления. (ч. 3). – Таганрог: Изд-во ТТИ ЮФУ, 2010. – 116 с.

¹Отрах В.В., ²Суконщиков А.А., ²Бахтенко Е.А.

Система кластерного анализа для выделения групп дебиторов в составе мультиагентной интеллектуальной системы поддержки принятия решений по оперативному рисковому ранжированию контрагентов

¹*Московский технологический институт
(Россия, Москва)*

²*Вологодский государственный университет
(Россия, Вологда)*

doi: 10.18411/lj-31-01-2017-1-08

idsp: 000001:lj-31-01-2017-1-08

Метод кластерного анализа применяется для разделения множества исследуемых признаков и объектов на кластеры или однородные группы в соответствующем понимании. Это значит, что задача классификации данных решается и выявляется соответствующая структура в ней. Управление дебиторской задолженностью является основной задачей финансового менеджмента любой организации. Возможность применения кластерного анализа в управлении дебиторской задолженностью является малоизученной и пока не нашла широкого отклика среди предпринимателей, поэтому рассматриваемая проблема является актуальной.

Кластерный анализ может быть использован циклически. В таком случае производится исследование до тех пор, пока не будут достигнуты нужные результаты. При этом каждый цикл, при этом использовании, может выдавать такую информацию, которая способна сильным образом изменить подходы и направление дальнейшего применения метода кластерного анализа. Данный процесс может иметь вид системы с обратной связью.

Большое преимущество кластерного анализа в том, что он производит разбиение объектов не по одному параметру, а по набору признаков в целом. Кроме того, кластерный анализ не накладывает ограничения на вид рассматриваемых объектов, в отличие от большинства экономико-математических методов, и позволяет рассмотреть, практически произвольной природы, большое количество исходных данных.

Кластерный анализ так же имеет свои недостатки и ограничения, а именно: количество и состав кластеров зависимы от разбиения выбираемых критериев. При сведении большого объема исходных данных к компактному виду возможны разные искажения, а также могут потеряться конкретные черты некоторых объектов, за счет изменения характеристик обобщенных значений различных параметров кластера [1].

Например, пусть Q включает n государств, одна из которых имеет ВВП на душу населения (B1), числом Н автомобилей на 1 тыс. человек (B2), потреблением электроэнергии на душу (B3), потреблением меди на душу (B4) и т.д. Тогда Y1 (вектор измерений) будет представлять собой набор перечисленных характеристик для первого государства, Y2 – для второго, Y3 для третьего, и т. д. В задаче необходимо разбить государства по уровню развития. Решением данной задачи будет – разбиение, удовлетворяющее определенному критерию оптимальности. Этот критерий, как правило, представляет собой функционал, который выражает уровни желательности различных группировок и разбиений и называется – целевой функцией. Здесь, самым трудным считается определить однородности объектов, задаваемых введением расстояний между

объектами u_i и u_j ($b(u_i, u_j)$). Однородными объектами будут в случае $b(u_i, u_j) \leq b_{пор}$, где $b_{пор}$ – заданное пороговое значение.

Основным моментом исследования является выбор расстояния (b), от этого расстояния зависят варианты разбиения в окончательном виде. Наиболее распространенными являются две процедуры: метод «ближнего соседа» и метод «дальнего соседа».

Метод «ближайшего соседа» основывается на наибольшей близости объектов по совокупности исследуемых признаков в различных кластерах. Эта близость должна группировать объекты вместе для формирования кластеров, в этом случае, результирующие кластеры могут быть представлены длинными «цепочками».

В методе «дальнего соседа» наоборот, расстояния между двумя любыми объектами в разных кластерах должно быть максимальным, тем самым определять расстояние между этими кластерами. Данный метод, как правило, работает хорошо в том случае, когда объекты различны. Если же кластеры схожи или их тип – «цепочечный», этот метод непригоден».

В задачах, решаемых методом кластерного анализа, достаточно часто применяют евклидово и хемингово расстояния.

Евклидово расстояние:

$$p_E(x_i, x_j) = . \quad (1)$$

С помощью этой формулы, берется в сравнении приближенность двух объектов по наибольшему числу признаков.

Хемингово расстояние:

$$p_H(x_i, x_j) = , \quad (2)$$

где k – количество признаков; i, j – признаки.

Формула используется как измерение различия объектов, задаваемых атрибутивными признаками.

Существует большое количество методов кластерного анализа. Опишем некоторые из них.

1) Объединение (древовидная кластеризация). Типичным примером такой кластеризации является иерархическое дерево (рис. 1).

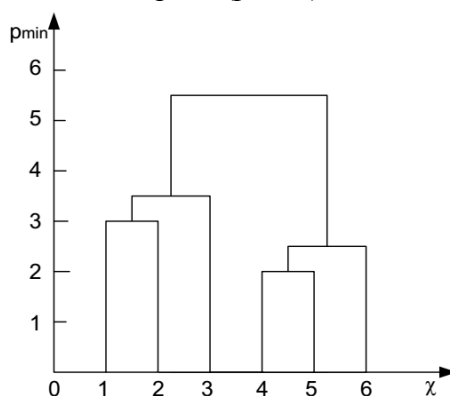


Рис. 1. Иерархическое дерево

На рисунке диаграмма начинается с каждого объекта в своем классе (в левой части рисунка). Предположим, что постепенно мы «понижаем» критерий об уникальности объекта, связываем вместе все большее и большее число объектов и агрегируем (объединяем) все больше число кластеров, состоящих из все сильнее различающихся элементов. В результате, на последнем шаге все объекты объединяются вместе.

На этом рисунке вертикальные оси означают расстояние объединения (в горизонтальных древовидных диаграммах горизонтальные оси означают расстояние

объединения). Следовательно, для каждого узла в графе (там, где формируется новый кластер) мы можем видеть величину расстояния, при котором соответствующие элементы связываются в новый единственный кластер.

Когда данные будут иметь понятную «структуру» в терминах кластеров объектов, сходных между собой, тогда эта структура, должна быть отражена в иерархическом дереве различными ветвями. В результате, если анализ методом объединения будет успешен, появится возможность обнаружить кластеры (ветви) и интерпретировать их.

2) Двухходовое объединение. Суть данного метода лучше показать на примере. Финансовому менеджеру необходимо собрать данные о различных характеристиках (переменные) финансового состояния контрагентов (наблюдений), которые имеют задолженность. Он может кластеризовать наблюдения (контрагентов) для определения кластеров контрагентов с похожим финансовым состоянием. В то же время, он может кластеризовать переменные (характеристики) для определения кластеров переменных, которые связаны со схожим финансовым состоянием. Двухходовое объединение позволяет провести кластеризацию в обоих направлениях. Так же, это объединение используется (достаточно редко) в обстоятельствах, когда ожидается, что и наблюдения и переменные одновременно могут вносить вклад в обнаружение осмысленных кластеров. Двухходовое объединение, как правило, наименее используемый метод.

Однако некоторые исследователи считают, что он предлагает мощное средство разведочного анализа данных.

3) Метод К-средних. Данный метод кластеризации сильно отличается от методов объединения описанных выше. Весь исходный набор примеров разбивается на k-классы таким образом, что максимизируется евклидово расстояние между классами и минимизируется евклидово расстояние между объектами внутри классов.

Выделение кластеров можно рассмотреть на примере.

Необходимо вычислить коэффициент ненадежности (табл. 1) проблемных дебиторов, который учитывает время, сумму заказа и сумму задолженности: $K_{нен.} = 1000 \times \text{ущерб} / \text{сумма заказа}$.

Таблица 1.

Коэффициент ненадежности проблемных дебиторов

	1-й дебитор	2-й дебитор	3-й дебитор	4-й дебитор	5-й дебитор	6-й дебитор
Коэффициент ненадежности	6	38	86	115	378	420

На основании коэффициента ненадежности проведем кластерный анализ проблемных контрагентов. Анализ будем проводить методом объединения (древовидная кластеризация). Выделяем расстояние r между коэффициентами по следующей формуле (1) (табл. 2).

Таблица 2.

Расстояние r между коэффициентами

	1	2	3	4	5	6
1	0	33	80	109	373	414
2	33	0	47	76	340	382
3	80	47	0	29	293	334
4	109	76	29	0	264	306
5	373	340	293	264	0	42
6	414	382	334	306	42	0

Считаем по принципу ближнего и дальнего соседа.

Принцип ближнего соседа.

	1	2	3	4	5	6
1	0	33	80	109	373	414
2		0	47	76	340	382
3			0	29	293	33
4				0	264	306
5					0	42
6						0

	1	2	3, 4	5	6
1	0	33	80	373	414
2		0	47	340	382
3, 4			0	264	306
5				0	42
6					0

	1, 2	3, 4	5	6
1, 2	0	47	340	382
3, 4		0	264	306
5			0	42
6				0

	1, 2	3, 4	5, 6
1, 2	0	47	340
3, 4		0	264
5, 6			0

	1, 2, 3, 4	5, 6
1, 2, 3, 4	0	264
5, 6		0

Рис. 3. Принцип ближнего соседа.

$P_{min} = P_{3,4} = 29$ ($S_1, S_2, S_{3,4}, S_5, S_6$); $P_{min} = P_{1,2} = 33$ ($S_{1,2}, S_{3,4}, S_5, S_6$); $P_{min} = P_{5,6} = 42$ ($S_{1,2}, S_{3,4}, S_5, S_6$); $P_{min} = P_{1,2,3,4} = 47$ ($S_{1,2,3,4}, S_5, S_6$).

В результате проведения кластерного анализа по принципу ближнего соседа получили два кластера (рис. 4).

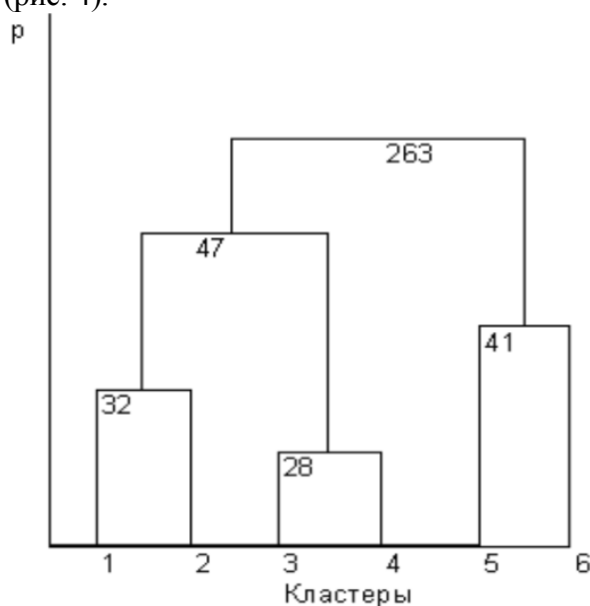


Рис. 4. Дендрограмма по принципу «ближнего соседа»

Принцип дальнего соседа.

	1	2	3	4	5	6
1	0	33	80	109	373	414
2		0	47	76	340	382
3			0	29	293	334
4				0	264	306
5					0	42
6						0

	1	2	3, 4	5	6
1	0	33	109	373	414
2		0	76	340	382
3, 4			0	293	334
5				0	42
6					0

	1, 2	3, 4	5	6
1, 2	0	109	373	414
3, 4		0	293	334
5			0	42
6				0

	1, 2	3, 4	5, 6
1, 2	0	109	414
3, 4		0	334
5, 6			0

	1, 2, 3, 4	5, 6
1, 2, 3, 4	0	414
5, 6		0

Рис. 5. Принцип дальнего соседа.

$P_{min} = P_{3,4} = 29$ ($S_1, S_2, S_{3,4}, S_5, S_6$); $P_{min} = P_{1,2} = 33$ ($S_{1,2}, S_{3,4}, S_5, S_6$); $P_{min} = P_{5,6} = 42$ ($S_{1,2}, S_{3,4}, S_5, S_6$); $P_{min} = P_{1,2,3,4} = 109$ ($S_{1,2,3,4}, S_5, S_6$).

Таким образом, при проведении кластерного анализа по принципу дальнего соседа получили два кластера (рис. 6).

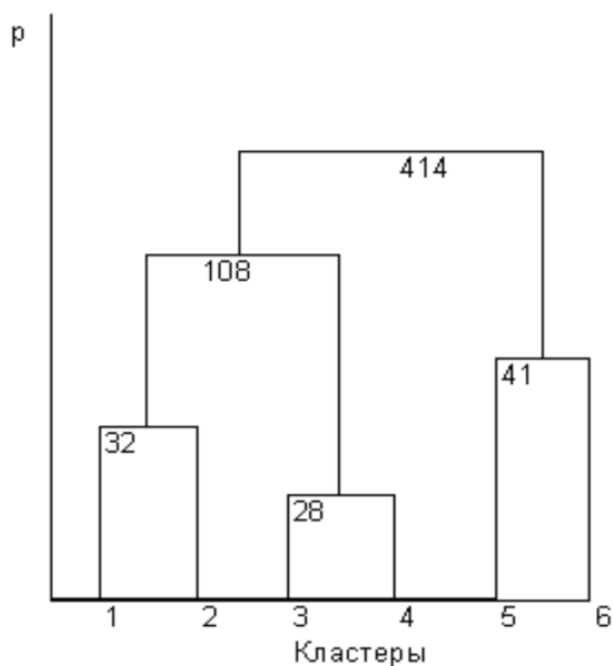


Рис. 6. Дендрограмма по принципу «дальнего соседа»

Проводим кластерный анализ только по проблемным контрагентам. У предприятия есть 14 покупателей, которые вовремя оплачивают платежи. Эти покупатели практически не наносят ущерб фирме. Общая сумма заказа этих контрагентов 3979 тыс. руб. Получается три кластера:

- 1) клиенты, которые расплачиваются немедленно за покупку («хорошие» дебиторы);
- 2) клиенты, у которых небольшой срок просрочки («ситуационные» дебиторы);
- 3) клиенты, у которых срок просрочки больше двух месяцев («плохие» дебиторы).

Всего покупателей за год 20. Далее рассчитывается ущерб, наносимый вторым и третьим кластерами.

Ущерб фирмы от задержки оплаты поставленных товаров покупателям (Tz_i) рассчитывается по формуле $Ущ.ф.= Ущ.и(Tz_i)$, при этом

$$Ущ.иC_y(Tz) =, \quad (3)$$

где p – годовая ставка банковских процентов по депозитам с учетом прироста инфляции:
 P – нормативный срок оплаты. инфл, $Tнс$

В итоге на основе кластерного анализа оптимизируем дебиторскую задолженность.

Таким образом, кластерный анализ позволяет выделять группы дебиторов (кластеры) основываясь на коэффициенте ненадежности, что позволяет выделять «хороших», «ситуационных» и «плохих» дебиторов и по-разному с ними работать. Применение кластерного анализа способствует оптимизации дебиторской задолженности организации и рекомендуется к применению в практической деятельности организаций различных сфер деятельности.

Список используемых источников информации

1. Суконщиков А.А., Дохтаева И.А. Современные методы интеллектуального анализа данных в СППР // Труды Международной научно-практической конференции «Информатизация инженерного образования» — ИНФОРИНО-2016 (Москва, 12—13 апреля 2016 г.). — М.: Издательский дом МЭИ, 2016. — 706 с.: ил.

**¹Отрах В.В., ²Суконщиков А.А., ²Кочкин Д.В., ²Бахтенко Е.А.
Система нечеткого вывода в составе мультиагентной интеллектуальной системы
поддержки принятия решений по оперативному рисковому ранжированию
контрагентов**

¹Московский технологический институт
(Россия, Москва)

²Вологодский государственный университет
(Россия, Вологда)

doi: 10.18411/lj-31-01-2017-1-09

idsp: 000001:lj-31-01-2017-1-09

Структура интеллектуальной СППР представлена на рисунке 1. В левой части схемы представлены источники данных о компании. Данные из всех источников объединяются для формирования профиля компании, который будет использован в системе нечеткого вывода для принятия решения о реструктуризации долга, продолжении сотрудничества, необходимости запроса дополнительных данных от контрагента и т.п. На основе профиля компании производится классификация. Для каждого класса компаний определяются дополнительные правила системы нечеткого вывода, а также веса для общих правил. Такой подход позволяет осуществлять принятие решений с учетом специфики отдельных компаний.

Для формирования базы правил необходимо определить множество входных лингвистических переменных и множество выходных лингвистических переменных .

Отдельная лингвистическая переменная задается следующим образом:

$$\beta_i = \langle \beta, T, X, G, M \rangle, \tag{1}$$

где β – наименование лингвистической переменной;

T – базовое терм-множество лингвистической переменной, каждый элемент которого представляет наименование отдельной нечеткой переменной а. .,

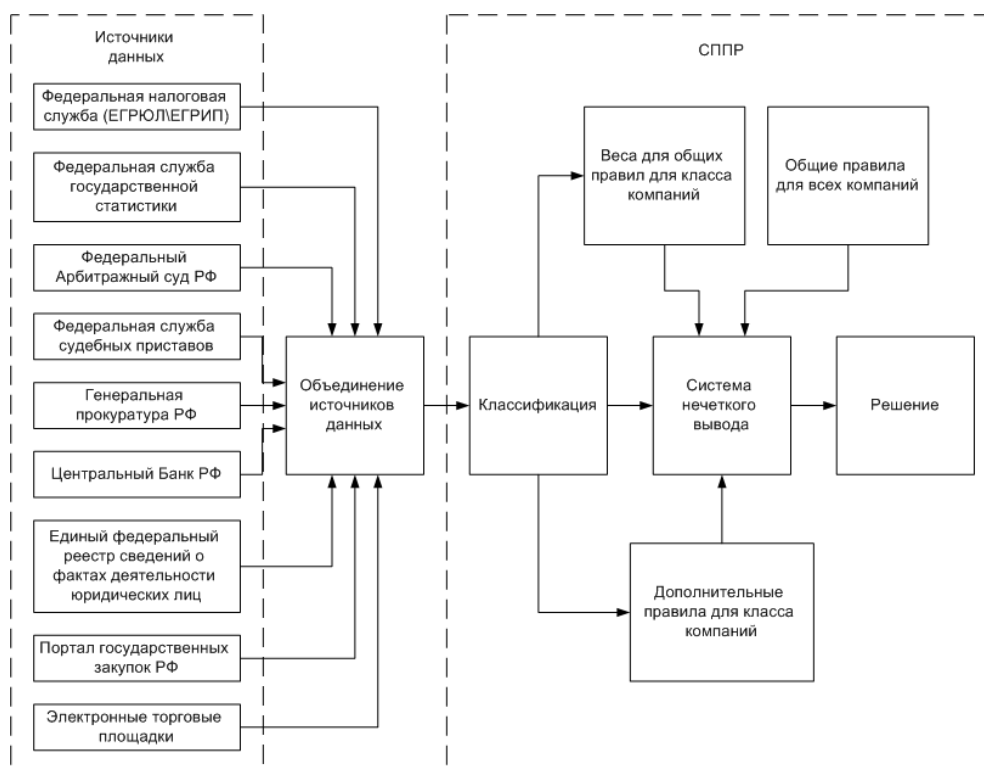


Рис. 1 - Схема СППР

X – область определения (универсум) нечетких переменных, входящих в ;

G – синтаксическая процедура, описывающая процесс образования из множества T новых, осмысленных в заданном контексте значений для ;

M – семантическая процедура, которая позволяет поставить в соответствие каждому значению, образованному с помощью G, некоторое осмысленное содержание посредством задания нечеткого множества, определенного на X.

Для задания функции принадлежности термов лингвистических переменных воспользуемся трапецеидальным нечетким интервалом (ТНИ).

$$A = \langle a, b, \alpha, \beta \rangle, \quad (2)$$

где a и b – нижнее и верхнее модальные значения ТНИ; α и β – левый и правый коэффициенты нечеткости ТНИ.

Функция принадлежности в виде ТНИ может быть задана следующим образом:

$$f(x; a, b, c, d) = \begin{cases} 0, & x \leq a \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ 1, & b \leq x \leq c \\ \frac{d-x}{d-c}, & c \leq x \leq d \\ 0, & x \geq d \end{cases} \dots\dots\dots(3)$$

База знаний интеллектуальной СППР может быть определена следующим образом:

$$R = \{R_1, R_2, \dots, R_m\} \mid m \in N, \quad (4)$$

где R_i – отдельное правило. Каждое правило базы знаний записывается следующим образом:

$$(R_i) : A \Rightarrow B, S, F, \quad (5)$$

где R_i – ядро правила; A и B – нечеткие высказывания, антецедент и консеквент правила; \Rightarrow – знак логического следования; S – метод определения количественного значения степени истинности заключения; F – коэффициент определенности правила. Коэффициент определенности правила может отсутствовать. В таком случае его значение принимается равным единице.

Рассмотрим следующие входные лингвистические переменные с базовыми терм множествами для системы нечеткого вывода:

1. Чистая прибыль (низкая, средняя, высокая);
2. Долг (низкий, средний, высокий);
3. Рост дохода (низкий, средний, высокий);
4. Спад числа сотрудников (низкий, средний, высокий).

Выходная лингвистическая переменная:

1. Риск (низкий, средний, высокий).

Терм множество для входной лингвистической переменной «Спад числа сотрудников» будет выглядеть следующим образом $T = \{A_{low}, A_{med}, A_{high}\}$; $A_{low} = \langle 0, 0, 8, 12 \rangle$, $A_{med} = \langle 8, 12, 20, 24 \rangle$, $A_{high} = \langle 20, 24, 36, 36 \rangle$; Функции принадлежности термов входной лингвистической переменной «Спад числа сотрудников» в виде трапецеидальных нечетких интервалов представлены на графике (рисунок 2).

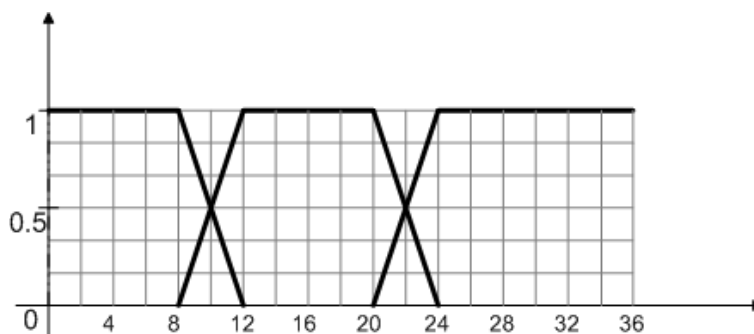


Рис. 2 - Функции принадлежности термов входной лингвистической переменной «Спад числа сотрудников»

Терм множество для выходной лингвистической переменной «Риск» будет выглядеть следующим образом: $T = \{A_{low}, A_{med}, A_{high}\}$; $A_{low} = \langle 0, 0, 20, 30 \rangle$, $A_{med} = \langle 20, 30, 50, 60 \rangle$, $A_{high} = \langle 50, 60, 100, 100 \rangle$; Функции принадлежности термов выходной лингвистической переменной «Риск» в виде трапецидальных нечетких интервалов представлены на графике (рисунок 3).

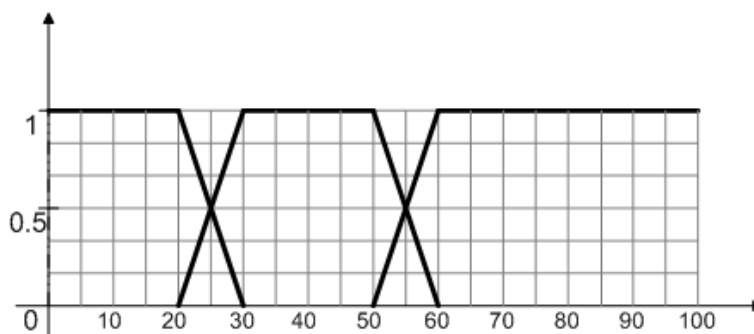


Рис. 3 - Функции принадлежности термов выходной лингвистической переменной «Риск»

В качестве возможных правил системы нечеткого вывода рассмотрим следующие:

- П1: ЕСЛИ Долг Высокий И ЧистаяПрибыль Низкая И РостДохода Низкий ТО Риск Высокий $F(П1)=1$
- П2: ЕСЛИ СпадЧислаСотрудников Высокий ТО Риск Высокий $F(П2)=0.5$
- П3: ЕСЛИ СпадЧислаСотрудников Средний ТО Риск Средний $F(П3)=1$

Рассмотрим работу системы нечеткого вывода для правил П2 и П3 с коэффициентами определенности $F(П2) = 0.5$ и $F(П3) = 1$. Четкое значение входной переменной СпадЧислаСотрудников примем равным 21. Таким образом истинность antecedента правила П2 будет равна 0.25, а степень истинности antecedента правила П3 будет равна 0.75.

После выполнения этапов активизации и аккумуляции будет получена функция принадлежности выходной лингвистической переменной «Риск» (Рисунок 4). При выполнении этапа активизации необходимо учитывать веса правил.

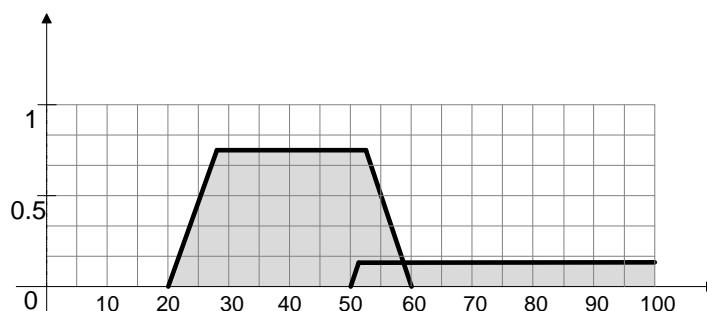


Рис. 4 - Функция принадлежности выходной лингвистической переменной Риск

Для получения четкого значения на основе функции принадлежности выходной лингвистической переменной можно воспользоваться одним из существующих методов дефаззификации, например: метод левого модельного значения, метод правого модельного значения, метод центра тяжести.

Для рассмотренных методов дефаззификации четкое значение выходной переменной «Риск» будет:

1. Метод левого модельного значения $y = 27,5$;
2. Метод правого модельного значения $y = 100$;
3. Метод центра тяжести $y = 46,8$.

Говоря о нечеткой логике в системах аналитики и управления, можно дать следующее определение системы нечеткого вывода. Система нечеткого вывода – это процесс получения нечетких заключений о требуемом управлении объектом на основе нечетких условий или предпосылок, представляющих собой информацию о текущем состоянии объекта. Этот процесс соединяет в себе все основные концепции теории нечетких множеств: функции принадлежности, лингвистические переменные, методы нечеткой импликации и т.п. База правил систем нечеткого вывода предназначена для формального представления эмпирических знаний экспертов в той или иной предметной области в форме нечетких продукционных правил. Таким образом, база нечетких продукционных правил системы нечеткого вывода – это система нечетких продукционных правил, отражающая знания экспертов о методах управления объектом в различных ситуациях, характере его функционирования в различных условиях и т.п., т.е. содержащая формализованные человеческие знания [1].

В заключение отметим, что задание жестких правил невозможно, так как нечеткость присутствует во входных данных по контрагентам, что приводит нас к использованию системы нечеткого вывода, которая позволяет осуществлять адаптивную оценку риска для класса компаний. Также более точный анализ компании-контрагента достигается за счет применения системы нечеткого вывода, содержащей знания экспертов, выраженные в правилах и весовых коэффициентах.

Список используемых источников информации

1. Суконщиков А.А., Дохтаева И.А. Применение интеллектуальных систем и средств нечеткой логики в автоматическом управлении // Научные исследования: от теории к практике: материалы IV Междунар. науч.-практ. конф. — Чебоксары: ЦНС «Интерактив плюс», 2015. — С. 188–189.

Свиридова И.В., Лебединская А.А., Внукова З.А., Кичигина А.К., Зайцева Т.В.
Разработка информационной подсистемы логистики на примере
ООО ТД «Малахов +»

*Белгородский государственный национальный исследовательский университет
(Россия, Белгород)*

doi: 10.18411/lj-31-01-2017-1-10

idsp: 000001:lj-31-01-2017-1-10

Логистика – это процесс создания оптимальной инфраструктуры движения товаров и услуг от поставщиков к потребителям [1].

Анализ работы логистов в России показывает, что наиболее профессионально важными для них являются знания и умения поиска информации, а также отбора и обработки приобретенных сведений, необходимых и достаточных для деятельности любого без логиста. Чтобы на должном и высоком уровне решать поставленные перед ним задачи, необходим соответствующий инструментарий и методология внедрения данного информационного обеспечения, в связи, с чем высока роль персонального

компьютера как инструмента, обеспечивающего эффективность работы с данными, и современных технологий хранения и поиска информации [3].

При разработке информационной подсистемы логистики целесообразнее всего нужно описать систему «КАК ЕСТЬ» и «КАК ДОЛЖНО БЫТЬ». На рисунке представлена контекстная диаграмма «КАК ЕСТЬ».

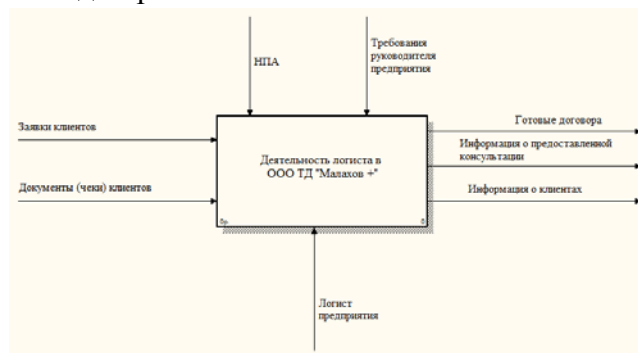


Рисунок 1 - Контекстная диаграмма «КАК ЕСТЬ»

Для решения данных проблем необходимо автоматизировать рабочее место логиста, так как это позволяет более удобно и быстро выполнять необходимые операции. Так же необходимо проводить действия по привлечению клиентов в ООО ТД «Малахов+», это позволит увеличить рост прибыли в компании.

Основной целью оптимизации и регламентации реинжиниринга является качественное улучшение деятельности компании, а также отдельных ее направлений за счет создания взаимосвязанного комплекта системообразующих документов, которые описывают наиболее значимые (ключевые) и часто реализуемые процессы [2].

На следующем рисунке представлена контекстная диаграмма «КАК ДОЛЖНО БЫТЬ» работы конфигурации «Деятельность логиста ООО ТД «Малахов +»».

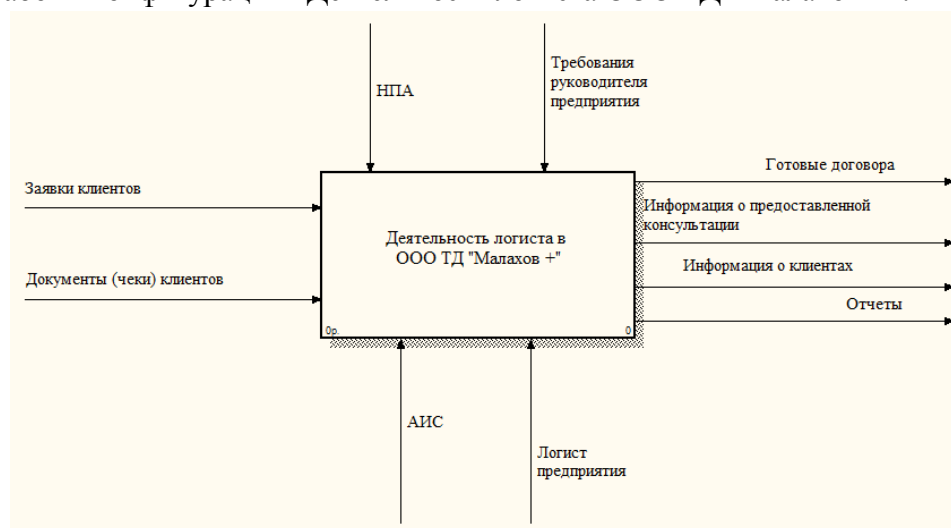


Рисунок 2 - Контекстная диаграмма «КАК ДОЛЖНО БЫТЬ»

В связи с удобством конвертации данных из электронных таблиц в систему 1С: Предприятие было принято решение разрабатывать собственную конфигурацию на встроенном языке программного продукта и создать специфические для ТД «Малахов+» виды отчетов.

Была разработана собственная конфигурация на платформе 1С: Предприятие 8.2 с использованием встроенного языка программирования 1С: Предприятие.

На основе спроектированной функциональной модели системы были созданы и настроены все необходимые объекты конфигурации. Система готова к использованию.

Был создан отчет, формирующий остатки товаров на предприятии в целом и по каждому ответственному сотруднику. Это позволяет эффективнее контролировать деятельность сотрудников.

Номенклатура	Балабина М. В.		Золотарев А.И.		Итого	
	Количество Остаток	Сумма Остаток	Количество Остаток	Сумма Остаток	Количество Остаток	Сумма Остаток
Колбаса			-198,000	12 177,00	-198,000	12 177,00
Конфеты	42,000	9 450,00	30,000	5 580,00	72,000	15 030,00
Молоко	26,000	1 920,00			26,000	1 920,00
Сахар			98,000	6 370,00	98,000	6 370,00
Итого	68,000	11 370,00	-70,000	24 127,00	-2,000	35 497,00

Рисунок 3 - Отчет по остаткам товаров

Еще один разработанный отчет показывает движение товаров на предприятии. Также он отображает движение товаров по отдельным сотрудникам. С его помощью намного упрощается процесс учета товаров.

Номенклатура	Балабина М. В.			Золотарев А.И.			Итого				
	Количество Начальный остаток	Количество Приход	Количество Расход	Количество Начальный остаток	Количество Приход	Количество Расход	Количество Начальный остаток	Количество Приход	Количество Расход		
Колбаса					123,000	321,000	-198,000		123,000	321,000	-198,000
Конфеты		45,000	3,000	42,000		30,000		75,000	3,000		72,000
Молоко		30,000	4,000	26,000				30,000	4,000		26,000
Сахар				98,000			98,000				98,000
Итого		75,000	7,000	68,000	251,000	321,000	-70,000	326,000	328,000		-2,000

Рисунок 4 - Отчет по товарам

Разработанная автоматизированная подсистема позволяет повысить оперативность и производительность логистов. За счет систематизации данных, система позволяет увеличить скорость и качество их работы. Разработанная система позволила синхронизировать используемые данные и сократить бумажные архивы. Данная конфигурация обладает интуитивно понятным интерфейсом, что упрощает работу с ней даже начинающим пользователям. Это ещё раз подчёркивает привлекательность этого решения ведения учёта.

Список используемых источников информации

1. Логистика и управление цепями поставок: учебник для академического бакалавриата / под ред. В. В. Щербакова. — М.: Юрайт, 2015. 582 с.
2. Радченко М.Г., Хрусталева Е.Ю. 1С Предприятие. Практическое пособие разработчика – М.: ООО «1С-Пабблишинг», 2009 – 857 с.
3. Хомичевская В. Переходим на 1С: Бухгалтерию. ИД "Питер – 1С: Библиотека – 2007, 496 с.

Смирнов А.В., Андрианов И.А., Суконщиков А.А., Бахтенко Е.А.
Математическая модель оптимизации доставки товаров автотранспортом на
разветвленной сети дорог для решения задач и кластеризации

*Вологодский государственный университет
(Россия, Вологда)*

doi: 10.18411/lj-31-01-2017-1-11

idsp: 000001:lj-31-01-2017-1-11

Для заданного региона обслуживания с помощью технологии ГИС предоставляется карта автомобильных дорог, на которой указаны пункты, соответствующие источникам (поставщикам) и приемникам (получателям или потребителям) грузов (товаров). Поставщику приписан парк автотранспорта, характеризующийся количеством автомобилей определенного типа и их массогабаритными параметрами. Поставщику поступают заявки от потребителей по количеству и ассортименту товаров. Каждый вид товара характеризуется массогабаритными параметрами. Ставится задача нахождения для заданного парка автотранспорта маршрутов развозов грузов от поставщика потребителям, обеспечивающее снижение суммарных затрат на перевозку товаров.

Заявки от потребителей, являющиеся входной информацией, поступают поставщику из Торговой системы.

Матрица взаимных расстояний между объектами развоза, также поступающая на вход системы, рассчитывается средствами ГИС. Рассматривается сетка дорог с большим количеством узлов – перекрестков, тупиков и точек обслуживания, через которые должны пройти маршруты движения транспорта. Сетке дорог ставится в соответствие ориентированный граф, вершинами которого являются узлы данной сетки, а ребрами – отрезки дорог между узлами (движение по дороге может быть односторонним). Каждому ребру приписывается длина – расстояние между соответствующими узлами. Ищется набор оптимальных маршрутов, начинающихся и заканчивающихся в заданных точках, и ограниченных некоторой функцией от длин ребер графа, которая может учитывать физическую длину маршрута (километраж), либо время движения транспорта, либо стоимостные характеристики маршрута движения.

Таким образом, расстояния между объектами задаются квадратной матрицей расстояний $A = [a(i, j)]$ размерности $n \times n$, где $a(i, j)$ – расстояние от пункта i до пункта j . Отметим, что, в общем случае, матрица расстояний не является симметричной (одностороннее движение, сложные транспортные развязки и т.д.).

Решение поставленной задачи развоза товаров осуществляется в два этапа. На первом этапе решается задача разбиения региона на компактные зоны обслуживания (группирование объектов-получателей для каждого маршрута). Эту задачу будем называть задачей кластеризации. На втором этапе решается задача нахождения оптимального по заданному критерию (суммарному расстоянию, времени, стоимости доставки) порядка объезда получателей для каждого маршрута. Эту задачу будем называть задачей маршрутизации. После решения этих двух задач формируются маршруты и расписания движения для всех автомобилей. Для каждого из этих двух этапов реализовано семейство алгоритмов, позволяющих получать результат в зависимости от размерности исходной задачи, вычислительных ресурсов и требований, предъявляемых к решению.

Задача кластеризации. Задача кластеризации реализуется набором методов (алгоритмов), каждый из которых осуществляет разбиения региона на компактные зоны обслуживания. Метод основан на выборе транзитивно ближайших сообщений.

При данном подходе каждому объекту сообщению ставится в соответствие пункт на карте. Пункты характеризуются расстояниями между собой.

Пусть $e = \max a(i, j)$ по всем i, j - максимальный элемент матрицы расстояний A .

Тогда матрица $B = [b(i, j)]$ размерности $n \times n$, где $b(i, j) = 1 - a(i, j) / e$ задает нечеткое отношение сходства. $0 \leq b(i, j) \leq 1$.

Транзитивное замыкание нечеткого отношения сходства задается матрицей $D = [d(i, j)]$ размерности $n \times n$. $0 \leq d(i, j) \leq 1$.

$$D = B \cup B^2 \cup B^3 \cup \dots \cup B^n, \quad (1)$$

где \cup – операция объединения нечетких отношений (MAX).

$B^2 = B \circ B$ представляет собой (Max - min)-композицию нечеткого отношения самого на себя.

$$b^2(x, z) = \text{MAX} [\text{MIN} (b(x, y), b(y, z))] \quad (2)$$

$B^{k+1} = B \circ B^k$ - (Max - min)-композиция нечетких отношений B^k и B .

$$b^{k+1}(x, z) = \text{MAX} [\text{MIN} (b^k(x, y), b(y, z))] \quad (3)$$

D – матрица задающая нечеткое отношение эквивалентности (подобия). Это отношение рефлексивно, симметрично и транзитивно. Согласно теореме о декомпозиции для отношения подобия, для каждого значения матрицы D (в порядке возрастания их значения) получаем транзитивно ближайшие сообщения (пункты).

Таким образом, в итоге получаем разбиение множества объектов на заранее заданное число компактных (транзитивно-ближайших) групп.

Раскраска графов.

Задача кластеризации может быть сведена к задаче раскраски вершин графа. Для этого строится граф несовместимости. Вершинам графа соответствуют объекты-потребители (пункты на карте), и две вершины - смежные, если соответствующие им объекты не могут находиться в одной группе. Причины несовместимости определяются различными факторами: большое расстояние между объектами, различие заявок – объектами заказаны разные товары, и т.п.

Ставится задача раскраски вершин такого графа несовместимости при различных условиях (ограничениях, критериях), среди которых:

- Минимальная раскраска графа (получение минимального числа компактных групп).
- Раскраска графа в заданное число красок (разбиение на группы, соответствующие парку автотранспорта).
- Раскраска графа в заданное число красок с ограничением на количество соцветных вершин (разбиение на группы, соответствующие парку автотранспорта и возможностям перевозки).

Для реализации метода используются оценки хроматического числа графа и спектр алгоритмов раскраски графов, учитывающих размерность решаемой задачи и отведенное на решение время.

Частотно-матричный метод. Подход к постановке задачи аналогичен предыдущему, но в качестве исходной модели рассматривается матрица инцидентий $Q = [q(i, j)]$. Столбцам матрицы соответствуют, буквы - объекты (пункты на карте), строкам – подмножество, определяемое словесным отношением S_i , $i=1, 2, \dots, n$, и $q(i, j) = 1$, если в i -е слово входит j -я буква, и $q(i, j) = 0$ – в противном случае.

Построение матрицы инцидентий осуществляется таким образом, что каждое словесное отношение, определяется выполнением условия совместности (несовместности) вхождения соответствующих объектов в одну группу.

На основе построенной матрицы определяется частотная матрица отношений $F = QT \times Q$, элементы которой $f(i, j)$ используются для целенаправленного нахождения ближайших объектов и включения их в одну группу. В основе данного подхода лежит аппарат дифференцирования дискретных моделей.

Алгоритм учитывает заявки, поступившие от потребителей.

Исходной моделью является матрица расстояний $A = [a(i, j)]$ размерности $n \times n$, где $a(i, j)$ – расстояние от пункта i до пункта j .

Ищется самая удаленная точка (пункт) и к ней последовательно добавляются ближайшие с учетом роста группы. В матрице A ищется строка, сумма элементов которой максимальна. Соответствующий ей пункт вносится в группу. i_1 – координата строки.

Ищется ближайший пункт к внесенному в решение – минимальный элемент матрицы A в столбце i_1 . Соответствующий ей пункт вносится в группу. i_2 – координата строки.

Ищется пункт ближайший к внесенным в решение – минимальная сумма элементов матрицы A в столбцах $i1$ и $i2$. Соответствующий ей пункт вносится в группу. $i3$ – координата строки.

И так далее. На очередном шаге k ищется пункт ближайший к внесенным в решение k пунктам $i1, i2, i3, \dots, ik$ – это минимальная сумма элементов матрицы A в указанных столбцах.

Процесс завершается, если величина заявки в очередной точке (пункте) $ik+1$ больше остатка $rest$, равного разности между максимально возможной величиной суммарных заявок (загрузка авто средства) и суммой заявок от всех включенных в решение пунктов $i1, i2, i3, \dots, ik$.

В зависимости от порядка организации перевозок пункт $ik+1$ может быть внесен в решение с числом выполненных заявок $rest$. В этом случае повышается общая загрузка авто средств, но пункт, будет посещаться неоднократно.

Смирнов А.В., Андрианов И.А., Суконщиков А.А., Бахтенко Е.А.
Математическая модель оптимизации доставки товаров автотранспортом на
разветвленной сети дорог для решения задачи маршрутизации

*Вологодский государственный университет
 (Россия, Вологда)*

doi: 10.18411/lj-31-01-2017-1-12
idsp: 000001:lj-31-01-2017-1-12

Предложено решение задачи развоза товаров, состоящее из двух этапов: разбиение региона на компактные зоны обслуживания (кластеризация); нахождение оптимального маршрута объезда пунктов развоза в рамках одного рейса (маршрутизация). Для каждого из этих этапов (кластеризации и маршрутизации) предложено семейство алгоритмов, обеспечивающих получение результата в зависимости от размерности решаемых задач, вычислительных ресурсов и требований, предъявляемых к решению.

Задача маршрутизации реализуется набором алгоритмов, каждый из которых осуществляет решение задачи коммивояжера. Коммивояжер (распространитель товаров) должен объехать всех получателей товаров внутри одной зоны обслуживания. Он выезжает из некоторого пункта и должен вернуться в него же в конце путешествия. Предполагается, что коммивояжер никогда не бывает дважды в одном пункте. Расстояния между получателями товаров задаются с помощью квадратной матрицы расстояний $C = [c(i, j)]$ размерности $k \times k$, где k – количество получателей товаров, $c(i, j)$ – расстояние от получателя i до получателя j . Отметим, что, в общем случае, матрица расстояний не является симметричной. Диагональные элементы матрицы расстояний равны нулю.

Задача коммивояжера состоит в таком объезде всех получателей, чтобы суммарное пройденное расстояние было минимальным. Следует выбрать один оптимальный маршрут из $(k-1)!$ возможных.

Если количество получателей невелико: ($k < 13$), то решение может быть получено с использованием простого метода перебора.

С ростом размерности задачи ($13 \leq k \leq 15$), целесообразно использовать метод ветвей и границ. Другое название этого же метода – метод поиска по дереву решений. Опишем основные шаги решения задачи коммивояжера этим методом. Вначале находится некоторое допустимое решение (допустимый маршрут). После этого на каждом шаге поиска оптимального решения множество всех оставшихся маршрутов разбивается на два подмножества. В первое подмножество включаются те маршруты, которые содержат некоторую выделенную дугу, во второе подмножество включаются те решения, которые эту выделенную дугу не содержат. Для каждого подмножества вычисляется нижняя граница стоимостей всех решений, вырастающих из этого подмножества. С

помощью найденных границ проводят дальнейшее разбиение подмножеств допустимых маршрутов. Алгоритм метода возвращается всякий раз, когда стоимость текущего частичного решения равняется или превосходит стоимость лучшего решения, найденного до сих пор. В конечном итоге остается один из оптимальных маршрутов. Отметим, что в некоторых задачах при использовании метода ветвей и границ объем вычислений может оказаться близким к объему вычислений методом перебора.

Для задач большей размерности ($k > 15$) метод ветвей и границ становится неприемлемым. В этом случае наиболее эффективным оказывается метод имитации отжига, основанный на аналогии с физическими эффектами, возникающими при охлаждении расплавленных металлов. При быстром охлаждении расплава последний обычно затвердевает, приобретая аморфные структуры расположения отдельных атомов. Эта неупорядоченная структура представляет собой одно из метастабильных состояний с локальным минимумом энергии. Однако перевести систему из расплава в кристаллическое состояние все же можно, если воспользоваться техникой отжига, заключающейся в очень медленном охлаждении системы с течением времени.

Алгоритм метода имитации отжига применительно к задаче коммивояжера состоит в построении последовательности укорачивающихся допустимых маршрутов с использованием локализованных мутаций, затрагивающих ровно два произвольно взятых

пункта объезда. Пусть $(i_1, i_2, \dots, i_m, i_{m+1}, \dots, i_p, \dots, i_k, i_1)$ - некоторый допустимый маршрут протяженности L_1 . Мутация, затрагивающая пункты i_m и i_p , заключается в том, что i_m и i_p меняются местами в последовательности объезда. Она приводит к

новому маршруту $(i_1, i_k, \dots, i_p, \dots, i_{m+1}, i_m, \dots, i_2, i_1)$ протяженности L_2 . Обозначим $\Delta L = L_2 - L_1$. В соответствии с алгоритмом метода имитации отжига результат мутации безоговорочно принимается, если она привела к сокращению протяженности маршрута: $\Delta L < 0$. Результат мутации принимается с некоторой вероятностью $Q = \exp(-\Delta L / H)$, если в результате мутации протяженность маршрута не уменьшилась: $\Delta L \geq 0$. Если мутация принята, то новый маршрут используется в качестве нового допустимого маршрута. Если же мутация отвергается, то допустимый маршрут остается прежним. Построение последовательности маршрутов продолжается до тех пор, пока протяженности маршрутов уменьшаются. Имитация отжига заключается в том, что значение параметра H , играющего роль температуры, от мутации к мутации уменьшается и стремится к нулю. Подбирая подходящую скорость уменьшения температуры, можно добиться того, что последовательность допустимых маршрутов будет стремиться к оптимальному маршруту.

Отметим, что метод имитации отжига позволяет найти лишь приближенное решение. В некоторых случаях достаточно эффективным оказывается приближенный, так называемый “хитро-жадный” метод. Этот метод является “усеченным” вариантом метода ветвей и границ. Отличие “хитро-жадного” метода от метода ветвей и границ состоит в том, что после проверок небольшого количества ветвей (жадность), содержащих наиболее короткие дуги графа (хитрость), вычисления прекращаются, и в качестве решения выбирается наилучшее решение, найденное к настоящему моменту.

В ряде случаев эффективными оказываются популярные в настоящее время адаптивно поисковые алгоритмы, основанные на эволюционных факторах получения решения, - генетические алгоритмы.

Смирнов А.В., Андрианов И.А., Суконщиков А.А., Бахтенко Е.А.
Математическая модель логистического управления циклом заказа с
использованием факторного анализа и механизмов прогнозирования

*Вологодский государственный университет
(Россия, Вологда)*

doi: 10.18411/lj-31-01-2017-1-13

idsp: 000001:lj-31-01-2017-1-13

Для увеличения надежности выполнения заказа за счет уменьшения цикла заказа предлагается внедрить в логистическую информационную систему математическую модель расчета коэффициента прогноза выполнения заказа $X_{\text{прогн}}$ коэффициента надежности выполнения заказа $Q_{\text{над}}$.

Модель реализуется следующим образом - информационной системе необходимо провести выборку факторов времени $X_{\text{п}}$ и факторов надежности $Q_{\text{п}}$, произвести анализ и выдать данные о планируемом (допустимом) времени выполнения заказа T .

В зависимости от требования, предъявляемого клиентом к выполнению заказа, производится расчет среднего времени выполнения заказа $X_{\text{ср}}$. Например, при выборе клиентом приоритета стоимости товара $S_{\text{зак}}$ информационная система определяет наиболее рациональное место отгрузки товара с учетом минимального $S_{\text{зак}}$, выбрав из базы данных случаи, по которым товар был доставлен из планируемого места поставки, и рассчитывает $X_{\text{ср}}$.

В общем виде уравнение расчета среднего времени заказа представлено в виде:

$$X_{\text{ср}} = \frac{\sum_{i=1}^n X_{\text{сли}}}{N_{\text{сл}}}, \quad ((1))$$

где $X_{\text{ср}}$ – среднее время выполнения заказа;
 $X_{\text{сли}}$ – время, выполнения i -й логистической операции;
 $N_{\text{сл}}$ – общее количество случаев.

В ряде работ для оценки изменения среднего квадратического отклонения σ_T используется формула:

$$\sigma_T = \sqrt{\sum_{i=1}^N \sigma_i^2 + 2 \sum_{i \leq j} r_{ij} \sigma_i \sigma_j}, \quad ((2))$$

где σ_i – среднее квадратическое отклонение времени выполнения i -й операции логистического цикла;

r_{ij} – коэффициент корреляции между i -й и j -й операциями цикла.

Однако формула (2) недостаточна для выявления причин, влияющих на отклонения планируемого времени выполнения заказа.

Для повышения обоснованности планируемого времени выполнения заказа предлагается формула:

$$T = X_{\text{ср}} \times X_{\text{прогн}} \times Q_{\text{над}}, \quad ((3))$$

где T – планируемое время выполнения заказа;
 $X_{\text{ср}}$ – среднее время выполнения заказа;
 $X_{\text{прогн}}$ – коэффициент прогноза выполнения заказа;
 $Q_{\text{над}}$ – коэффициент надежности выполнения заказа.

На основании анализа времени выполнения заказов и влияющих на них факторов, полученных из информационной базы данных, применяя метод корреляционно-

регрессионного анализа, автором были выявлены факторы, воздействующие на цикл заказа.

Для вычисления $X_{\text{прогн}}$ и $Q_{\text{над}}$ представлены ключевые факторы времени цикла заказа в розничной торговле КТ (рис. 1).

В результате факторного анализа получено уравнение для расчета $X_{\text{прогн}}$, имеющее вид:

$$X_{\text{прогн}} = 1,0599 + 2,459 \times X1 + 1,641 \times X2 + 1,2166 \times X3 + 0,832 \times X4 + 0,335 \times X5 + 0,9988 \times X6 + 0,754 \times X7 + 1,448 \times X8 + 1,9404 \times X9 + 1,641 \times X11 + 1,2166 \times X12 + 0,832 \times X13 + 0,335 \times X14 + 0,9988 \times X15 \quad (4)$$

Факторы $X10$ и $X16$ математической моделью не учитываются, так как они не оказывают значительного влияния на результативный признак.

Факторы	Описание факторов времени	Факторы	Описание факторов надежности
X1	Сезонность	Количество брака на 1000 ед. товаров	
Время, затрачиваемое клиентом на подачу заявки			
X2	используя интернет-магазин	X17	категории «А»
X3	используя терминал	X18	категории «В»
X4	при помощи консультанта	X19	категории «С»
Время, затрачиваемое на обработку заявки		X20	ошибка менеджера
		X21	возврат товаров
X5	используя интернет-магазин	Задержки поставки заказа	
X6	используя терминал		
X7	при помощи консультанта		
Время, затрачиваемое клиентом на оплату		X22	не более 3 часов
		X23	более 3 часов
X8	с использованием кредита	X24	более 24 часов
X9	наличными средствами		
X10	электронным способом	X25	сбой информационной системы
Время, затрачиваемое на комплектацию		X26	ошибки поставщиков
X11	товаров категории «А»		
X12	товаров категории «В»		
X13	товаров категории «С»		
Время, затрачиваемое на доставку заказов клиенту			
X14	поставщиками через РЦ		
X15	распределительными центрами		
X16	транспортной службой магазина		

Рис. 1. - Ключевые факторы времени и факторы надежности в цикле заказа розничной торговли

Для $Q_{\text{над}}$ уравнение имеет вид:

$$Q_{над} = 1,11 + (-9 \times X17 - 39 \times X18 - 17 \times X19 - 16,3 \times X21 - 9 \times X22 - 39 \times X23 - 17 \times X24 - 16,3 \times X25) / 100 \quad ((5))$$

Факторы X20 и X26 математической моделью не учитываются, так как они также не оказывают значительного влияния на результативный признак при регрессионном анализе.

В целях рационализации выполнения логистических операций по реализации заказов предлагаются методические рекомендации по определению, на основе статистического моделирования, допустимого времени выполнения заказа, дифференцированного по каждой операции логистического процесса заказа, и использованию допустимого (планируемого) времени выполнения заказа (Т), с которым сравнивается фактическое время.

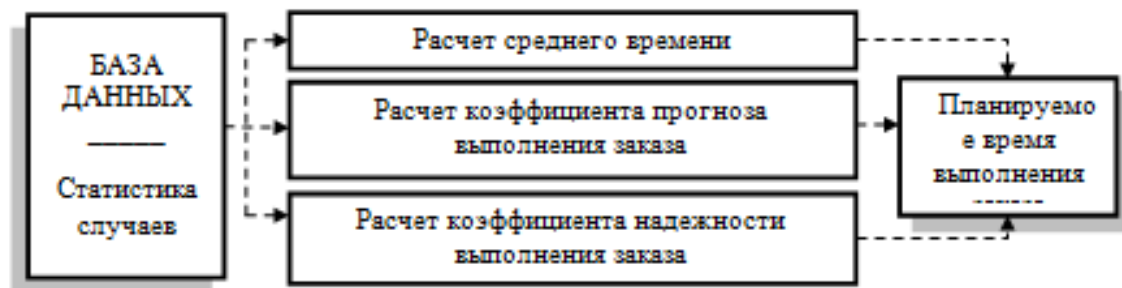


Рис. 2. - Схема расчета планируемого времени выполнения заказа

При выявленных отклонениях от допустимого времени выполнения заказа определяется зона ответственности, по вине которой произошло это отклонение, выявляются причины и принимаются управленческие решения. Например, для доставки товара на склад одного из РЦ в среднем уходит 2,5 часа. Фактически для выполнения данной операции потребовалось 3,6 часа. В результате статистического моделирования и расчета установлено допустимое время 4 часа. При превышении времени информационная система устанавливает, по вине какой зоны ответственности это произошло, то есть, на какой логистической операции. Информация поступает в систему логистического менеджмента компании, который выявляет причины отклонений и принимает меры по повышению надежности выполнения заказа.

Смирнов А.В., Андрианов И.А., Суконщиков А.А., Бахтенко Е.А.

Разработка модели адаптивного управления исполнением заказа на основе вероятностно-статистического подхода в логистической концепции «точно-в-срок»

*Вологодский государственный университет
(Россия, Вологда)*

doi: 10.18411/lj-31-01-2017-1-14

idsp: 000001:lj-31-01-2017-1-14

В литературе по логистике понятие «точно-в-срок» («точно-вовремя») рассматривается применительно к логистическому циклу, который является одним из основных объектов интегрированной логистики. Поскольку временные интервалы выполнения отдельных операций логистического цикла являются случайными величинами, то и продолжительность всего цикла, которая, как правило, представляет сумму времени выполнения отдельных элементов (операций), является случайной величиной, подчиняющейся определенному закону распределения.

Зная продолжительность логистического цикла, можно найти время выполнения цикла заказа в точно обозначенные сроки. Заказчик может выдвигать требование доставить товары в определенное (точное) время или же задать время доставки с учетом небольшого отклонения, которое он считает допустимым. Если время выполнения заказа «точно-в-срок» задано каким-то определенным значением времени, время цикла заказа является верхней доверительной границей времени и может быть рассчитано по формуле:

$$T_{TB}^g = T_n + \bar{T} + x_p \sigma_T, \quad (1)$$

где T_n – время начала выполнения логистического цикла;

\bar{T} – среднее значение времени логистического цикла;

σ_T – среднее квадратическое отклонение времени цикла.

Если время выполнения заказа «точно-в-срок» задано не только ориентировочным значением, но и некоторым отклонением от него или интервалом времени, важно оценить не только верхнюю, но и нижнюю границу времени выполнения заказа

$$T_{TB}^H = T_n + \bar{T} - x_p \sigma_T. \quad (2)$$

Для снижения риска невыполнения заказа в договорные сроки важно уметь управлять процедурами заказа, в частности, выбирать наилучший по времени вариант выполнения операций логистического цикла. Под управлением процедурами цикла заказа понимается в данном случае корректировка составляющих цикла в ходе организации и выполнения заказа потребителя, замена некоторых составляющих или ускоренное их осуществление. Например, можно оперативно заменять вид транспорта, перевозчика, маршрут или приобрести компоненты заказа у другого поставщика, но при этом не нарушить требование относительно выполнения всего цикла заказа в точные сроки.

На рис. 1 показано, что выполнение заказа в заданные сроки (линия 1) возможно с вероятностью P_1 , меньшей заданной надежности. Для выполнения требования заказчика необходимо приблизить заданное время прибытия заказа к потребителю и гарантированное с вероятностью 0,9 время доставки точно в срок. Из анализа вышеприведенных зависимостей для определения времени выполнения заказа следует, что выполнение условия «точно в срок» может быть достигнуто различными способами, которые позволят приблизить заданное время выполнения заказа и гарантированное с вероятностью 0,9 время точно в срок.

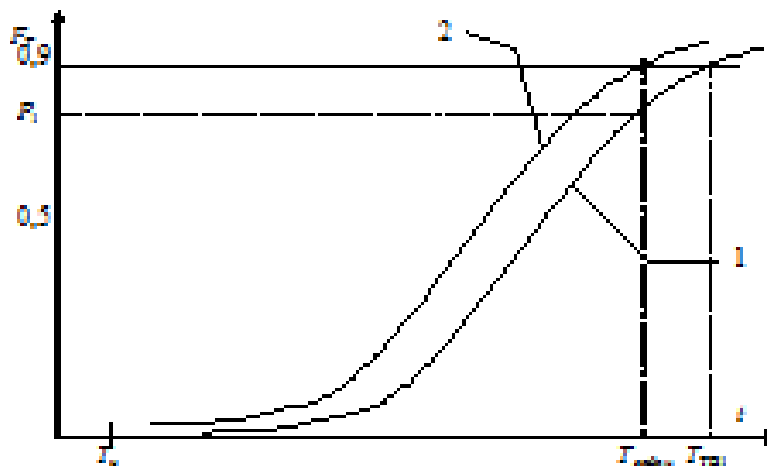


Рис. 1. - Зависимость функции распределения времени выполнения заказа от изменения среднего времени цикла

90-процентным временем выполнения заказа. При этом среднее время выполнения заказа остается прежним. Однако чаще всего изменяются среднее квадратическое отклонение и среднее значение времени (рис. 3, линия 4). Например, таможенное оформление товара при помощи таможенного брокера займет в среднем меньше времени, чем самостоятельное оформление. Экспертным путем и на основании обследований маршрутов установлено, что колеблемость значений времени оформления перевозчиком существенно больше по сравнению с вариантом оформления с использованием брокера. Однако брокер оказывает услуги на возмездной основе, что необходимо учитывать при принятии решений.

В-четвертых, необходимо учесть, что продолжительности операций логистического цикла могут быть связаны между собой, и задержка при выполнении одной операции, например, при комплектовании заказа, потребует уменьшения времени другой операции – доставки, то есть возникает корреляционная связь. Учет свойств обратной (отрицательной) корреляции времени отдельных операций логистического цикла при условии, что это не приведет к росту остальных t_{ij} , может изменить среднее квадратическое отклонение времени выполнения заказа.

Как уже отмечалось, в ряде случаев заказчик требует доставить необходимые материалы, сырье и т.п. в некотором интервале времени ($\bar{T} \pm \delta$), тем самым он фактически определяет не только среднее, но и среднее квадратическое отклонение времени выполнения заказа. Это ведет к появлению вариантов выполнения заказа раньше времени, заданного потребителем. В таком случае, необходимо обеспечить равенство среднего времени выполнения заказа с заданным средним значением времени и применить варианты управленческих решений, направленные на сближение расчетных и договорных сроков доставки грузов. Однако равенство расчетного и заданного среднего времени выполнения заказа не обеспечивает надежности, поскольку среднее квадратическое отклонение времени доставки может быть слишком велико. Это ведет к появлению фактически возможных вариантов выполнения заказа раньше времени, заданного потребителем (заказчик не разгружает прибывшие товары, транспортное средство вместе с водителем простаивает в ожидании разгрузки), и позднее сроков, указанных в договоре. Это ведет к необходимости уплатить штраф за срыв сроков поставки, отказу клиента в приеме заказа и к другим последствиям.

Во всех случаях отклонение сроков выполнения заказа от заданного интервала ведет к затратам поставщика, поэтому при управлении процедурами заказов необходимо на стадии проектирования логистического цикла подобрать такие варианты выполнения операций цикла, чтобы сумма верхних (нижних) оценок доверительных интервалов времени выполнения заказа давала результат, принадлежащий интервалу времени, заданному заказчиком. Рассмотренные варианты решений могут быть применены не только при проектировании цепи поставок, в которой будут выполняться логистические циклы, но и при управлении составляющими конкретными заказами.

Учитывая характер решаемых задач при управлении циклом заказа и случайность временных характеристик элементов цикла, основным методом проектирования времени выполнения заказа может являться имитационное моделирование.

Смирнов А.В., Андрианов И.А., Суконщиков А.А., Бахтенко Е.А.
Разработка иерархической классификации оптимизационных моделей
управления цепями поставок

Вологодский государственный университет
(Россия, Вологда)

doi: 10.18411/lj-31-01-2017-1-15

idsp: 000001:lj-31-01-2017-1-15

Рассмотрим вопрос разработки иерархической классификации оптимизационных моделей управления цепями поставок, которая позволит уточнить области применения моделей, а также послужит концептуальной основой для выбора метода и средства решения конкретных задач планирования бизнес-процессов в цепях поставок. Вопросы классификации моделей и методов теории логистики рассматриваются в ряде работ, однако, многие попытки классификации и систематизации моделей и методов, возможных для использования в логистике, а следовательно, и определение сфер их применения, выглядят довольно хаотично. Например, в модель включаются разные по измеримости массивы информации, совершенно несопоставимые друг с другом; переменные модели очень условны; отсутствуют ограничения, которые и должны определить решаемость модели, ее реалистичность.

Проведенные исследования используемых в логистике экономико-математических моделей и методов позволяют сделать вывод о том, что необходимо уточнение существующих классификаций. В результате предложена иерархическая классификация моделей управления цепями поставок, представленная на рис. 1.

На верхнем уровне иерархии все модели предлагается делить «по бизнес-функциям» на два типа: модели учета издержек (транзакционные) и модели операций (аналитические).

Дальнейшая классификация разрабатывалась только для моделей операций, которые на втором уровне иерархии предлагается делить «по степени определенности» на два класса: детерминированные и неопределенные.

На третьем уровне иерархии предлагается делить модели на группы «по математическим свойствам». При этом детерминированные модели будут подразделяться на линейные, нелинейные и многоцелевые модели, а неопределенные модели – на модели принятия решений, игровые модели, модели управления запасами, модели прогнозирования, модели систем массового обслуживания и имитационные модели.

На четвертом уровне предлагается делить все оптимизационные модели «по охватываемому временному интервалу» на статические (однопериодные) и динамические (многопериодные).

На пятом уровне предлагается деление всех статических и динамических моделей «по виду переменных» на четыре подгруппы: с непрерывными переменными, с целочисленными переменными, с булевыми переменными, с переменными смешанного типа, т.е. одна часть переменных является непрерывными, а другая часть целочисленными или булевыми. Такие модели в специальной литературе называют также моделями смешанного программирования.

На шестом уровне предлагается делить все оптимизационные модели (и статические, и динамические) «по бизнес-процессам» на следующие группы: планирования снабжения, производственного планирования, планирования распределения, транспортные и интегрированные модели цепей поставок, в частности, транспортно-складская модель, производственно-транспортно-складская модель, интегрированная модель цепи поставок и маркетинга, интегрированная модель цепи поставок и финансового планирования и др.

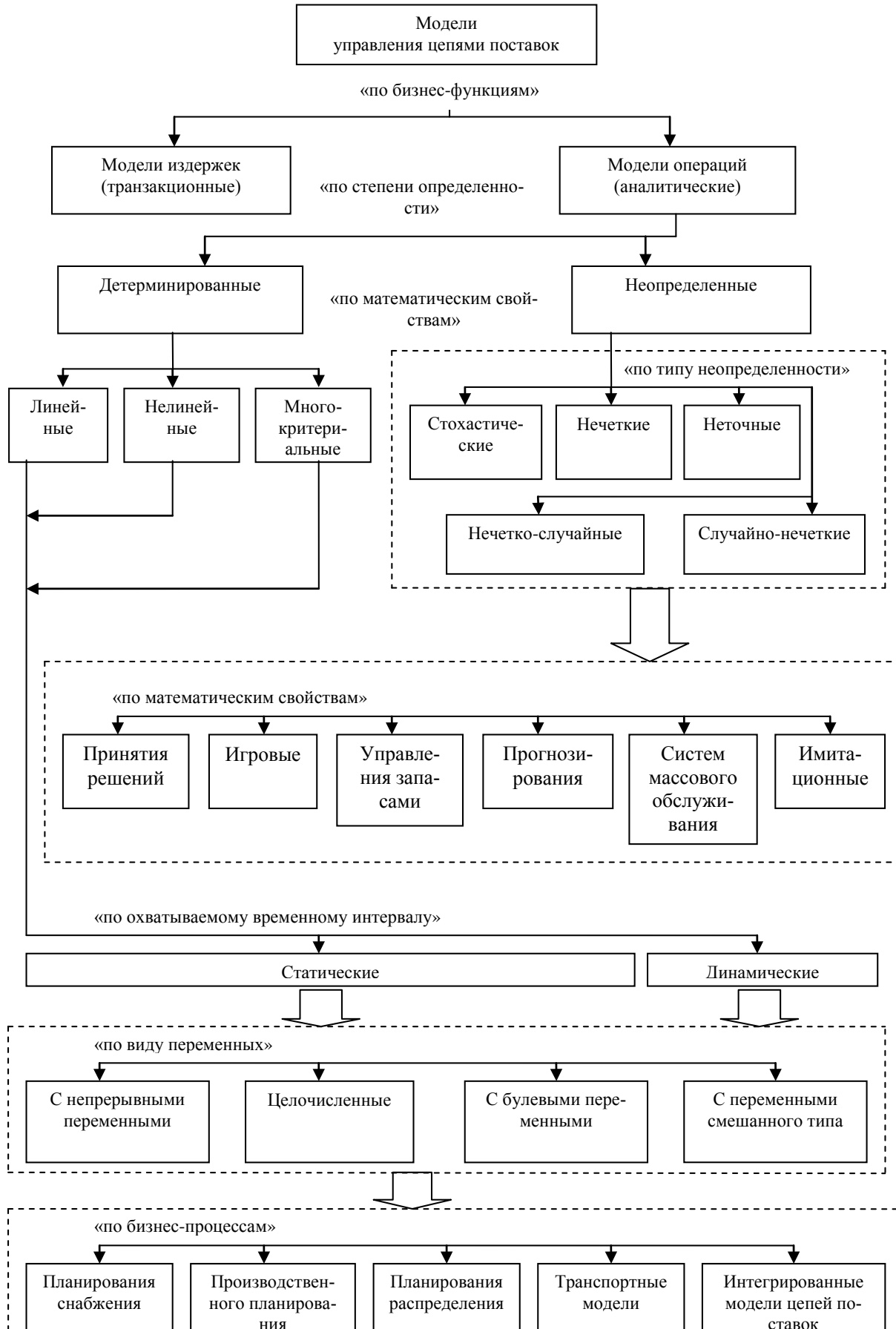


Рис. 1. - Классификация оптимизационных моделей управления цепями поставок

В настоящее время появились новые исследования в области теории и практики проектирования нечетких систем и неопределенного программирования. В данных работах, в частности, наряду со стохастическим программированием рассматриваются также нечеткое программирование, неточное программирование, нечетко-случайное программирование, случайно-нечеткое программирование и другие виды неопределенного программирования.

К построению различных моделей неопределенного программирования приводит различие в математических свойствах величин, отражающих неопределенность. Этот новый взгляд на классификацию моделей «по степени определенности» нашел отражение на рис. 1.

Классификация неопределенных моделей не представлена рис. 1, но она, по нашему мнению, должна соответствовать общепринятой классификации. Например, модели принятия решений, подразделяются «по предположениям о поведении природы» на три подгруппы: модели принятия решений в условиях определенности, модели принятия решений в условиях риска и модели принятия решений в условиях неопределенности. Модели прогнозирования подразделяются «по категориям» на причинно-следственные модели и модели временных рядов.

По нашему мнению, основных свойств базовых компонентов модели (переменных, ограничений, целевой функции) оказывается вполне достаточно для классификации большинства конкретных моделей оптимизации цепей поставок, встречающихся на практике. Представленная выше иерархическая классификация моделей управления цепями поставок, позволяет глубже понять, где применяются и как используются те или иные модели, а также служит концептуальной основой для адекватного выбора метода и средства решения конкретных задач того или иного класса.

Научное издание

Тенденции развития науки и образования

Сборник научных трудов, по материалам
XXII международной научно-практической конференции
31 января 2017 г.
Часть 1



SPLN 001-000001-0093-5B

Подписано в печать 11.02.2017. Тираж 400 экз.
Формат.60x841/16. Объем уч.-изд. л.2.76
Бумага офсетная. Печать оперативная.
Отпечатано в типографии НИЦ «Л-Журнал»
Главный редактор: Иванов Владислав Вячеславович