

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ,
МОЛОДЕЖИ И СПОРТА УКРАИНЫ

ХАРЬКОВСКИЙ НАЦИОНАЛЬНЫЙ
УНИВЕРСИТЕТ РАДИОЭЛЕКТРОНИКИ

ISSN 0135-1710

АВТОМАТИЗИРОВАННЫЕ СИСТЕМЫ УПРАВЛЕНИЯ И ПРИБОРЫ АВТОМАТИКИ

**Всеукраинский межведомственный
научно-технический сборник**

Основан в 1965 г.

Выпуск 161

Харьков
2012

СОДЕРЖАНИЕ

ХАХАНОВА И.В. КВАНТОВЫЙ ПРОЦЕССОР ОПТИМАЛЬНОГО ПОКРЫТИЯ.....	4
ХАХАНОВ В.И., ANDERS CARLSSON, ЧУМАЧЕНКО С.В., БУТЕНКО С.А. МОДЕЛИ УПРАВЛЕНИЯ УЯЗВИМОСТЬЮ.....	10
МИЗЬ В.А., ХАХАНОВА А.В. АНАЛИЗ СИСТЕМ АВТОМАТИЗИРОВАННОГО МОНИТОРИНГА АВТОМОБИЛЬНОГО ТРАНСПОРТА И УПРАВЛЕНИЯ ДОРОЖНЫМ ДВИЖЕНИЕМ.....	25
БОГОМОЛОВ В.А., НЕВЛЕВА С.Н., РАЗНИЦЫН И.Л., СИДОРОВ М.В. ЧИСЛЕННЫЙ АНАЛИЗ НАПРЯЖЕННО-ДЕФОРМИРОВАННОГО СОСТОЯНИЯ СЛОЯ ДОРОЖНОЙ ОДЕЖДЫ КАК ЛИНЕЙНОЙ И ГЕОМЕТРИЧЕСКИ-НЕЛИНЕЙНОЙ ВЯЗКОУПРУГОЙ СРЕДЫ НА ОСНОВАНИИ РЕОЛОГИЧЕСКОЙ ОДНОЭЛЕМЕНТНОЙ МОДЕЛИ КЕЛЬВИНА.....	31
ЛЯШЕНКО С.А., ЛЯШЕНКО А.С. ПОСТРОЕНИЕ ЛИНЕАРИЗИРОВАННЫХ МАТЕМАТИЧЕСКИХ МОДЕЛЕЙ РАБОТЫ ВАКУУМ-АППАРАТОВ КРИСТАЛЛИЗАЦИОННОГО ОТДЕЛЕНИЯ САХАРНОГО ЗАВОДА.....	38
БОЖИНСКИЙ И.А. ОБ ОДНОМ ПОДХОДЕ К МОДУЛЬНОМУ ПРОЕКТИРОВАНИЮ ИНФОРМАЦИОННЫХ СИСТЕМ.....	41
КОРАБЛЁВ Н.М., ФОМИЧЁВА А.А. ПОВЫШЕНИЕ ЭФФЕКТИВНОСТИ ИММУННЫХ МЕТОДОВ КЛАССИФИКАЦИИ ОБЪЕКТОВ С ПОМОЩЬЮ ЦЕЛЕВОГО КЛОНАЛЬНОГО ОТБОРА.....	45
БЕРЗЛЕВ А.Ю. РАЗРАБОТКА КОМБИНИРОВАННЫХ МОДЕЛЕЙ ПРОГНОЗИРОВАНИЯ С КЛАСТЕРИЗАЦИЕЙ ВРЕМЕННЫХ РЯДОВ ПО МЕТОДУ БЛИЖАЙШЕГО СОСЕДА.....	51
НОВИКОВ Ю.С. СТРУКТУРИЗАЦИЯ СОСТАВНЫХ ОБЪЕКТОВ ПРИ ФОРМИРОВАНИИ ПРОЦЕССНОГО ПРЕДСТАВЛЕНИЯ ЗНАНИЙ.....	59
ЛИТВИН В.В., ГОПЯК М.Я., ДЕМЧУК А.Б. МЕТОД АВТОМАТИЗОВАНОЇ РОЗБУДОВИ ТА ОЦІНЮВАННЯ ЯКОСТІ ОНТОЛОГІЙ БАЗ ЗНАНЬ.....	62
ГВОЗДИНСКИЙ А.Н., ЯНОВ Д.М. ИССЛЕДОВАНИЕ ЗАДАЧ ПРИНЯТИЯ РЕШЕНИЙ В СИСТЕМАХ УПРАВЛЕНИЯ ОГРАНИЧЕННЫМИ РЕСУРСАМИ.....	69
ГУРИН В.Н., ФИРСОВ А.Г., ГУРИН Д.В. МАТЕМАТИЧЕСКИЕ МОДЕЛИ ТЕХНОЛОГИЧЕСКОГО ПРОЦЕССА НАПЫЛЕНИЯ НАНОСТРУКТУРИРОВАННЫХ ДИЭЛЕКТРИЧЕСКИХ ПЛЕНОК, ПОЛУЧЕННЫХ МЕТОДОМ ИОННО-ПЛАЗМЕННОГО РАСПЫЛЕНИЯ.....	74
САКАЛО Е.С. ИССЛЕДОВАНИЕ И ПОСТРОЕНИЕ ОБЛАЧНОЙ МОДЕЛИ УСЛУГ В CLOUD-СРЕДЕ.....	77
ГУБНИЦКАЯ Ю.С. ИНФОРМАЦИОННЫЕ МОДЕЛИ ДАННЫХ И КРИТЕРИИ ОЦЕНКИ КАЧЕСТВА КОМПОНОВКИ СТАТЕЙ НА ПОЛОСЕ ПРИ ДОПЕЧАТНОЙ ПОДГОТОВКЕ ИЗДАНИЙ.....	81
ІВЛЄВА С.М. ІНТЕРАКТИВНА СИСТЕМА МОДЕЛЮВАННЯ ТА ОПТИМІЗАЦІЇ РЕЖИМІВ РОБОТИ КОМПРЕСОРНОГО ЦЕХУ З УРАХУВАННЯМ РОБОТИ СИСТЕМИ АВТОМАТИЧНОГО УПРАВЛІННЯ.....	89
ЧАЛЫЙ С.Ф., БОГАТОВ Е.О., МЕЛЕШКО Д.Г. ТЕХНОЛОГИЯ ПРЕДВАРИТЕЛЬНОЙ СТРУКТУРИЗАЦИИ ЖУРНАЛОВ РЕГИСТРАЦИИ СОБЫТИЙ СЛАБОСТРУКТУРИРОВАННЫХ БИЗНЕС-ПРОЦЕССОВ.....	98
ЧАЛЫЙ С.Ф., БУЦУКИНА И.Б. МОДЕЛЬ БИЗНЕС-ПРОЦЕССА С ИЗМЕНЯЕМОЙ НА ОСНОВЕ ПРАВИЛ СТРУКТУРОЙ.....	103
ЧАЛЫЙ С.Ф., АЛЬШЕЙХ АЛИ ДЖАМИЛЬ. СЕРВИС-ОРИЕНТИРОВАННАЯ МОДЕЛЬ БИЗНЕС-ПРОЦЕССА С ИЗМЕНЯЕМОЙ СТРУКТУРОЙ.....	106
ГВОЗДИНСКИЙ А.Н., ЯКИМОВА Н.А., ГУБИН В.А. БИНАРНЫЕ ПРЕДИКАТЫ ПРИ ОПИСАНИИ БУЛЕВЫХ ЛОГИЧЕСКИХ ПРОСТРАНСТВ.....	108
НЕВЛЕВ Е.С. О ВЫБОРЕ ЗАКОНА РАСПРЕДЕЛЕНИЯ ПРОДОЛЖИТЕЛЬНОСТИ ПЕРЕДАЧИ ПАКЕТИРОВАННЫХ ДАННЫХ В КОРПОРАТИВНЫХ КОМПЬЮТЕРНЫХ СЕТЯХ.....	113
КИРИЧЕНКО Л.О., РАДИВИЛОВА Т.А., КАЙАЛИ Э. РАСЧЕТ СТОИМОСТИ МАРШРУТИЗАЦИИ В СЕТИ MPLS С УЧЕТОМ ФРАКТАЛЬНЫХ СВОЙСТВ ТРАФИКА.....	116
ОКСАНИЧ А.П., ПРИТЧИН С.Э., ТЕРБАН В.А. УСОВЕРШЕНСТВОВАНИЕ МЕТОДА ИЗМЕРЕНИЯ ОСТАТОЧНЫХ НАПРЯЖЕНИЙ В ПОДЛЮЖКАХ АРСЕНИДА ГАЛЛИЯ.....	122
РЕФЕРАТЫ.....	129
ПРАВИЛА ОФОРМЛЕНИЯ СТАТЕЙ ДЛЯ АВТОРОВ НАУЧНО-ТЕХНИЧЕСКОГО СБОРНИКА.....	136

КВАНТОВЫЙ ПРОЦЕССОР ОПТИМАЛЬНОГО ПОКРЫТИЯ

Предлагаются кубитные (квантовые) структуры данных и вычислительных процессов для существенного повышения быстродействия при решении задач поиска, распознавания, принятия решений, дискретной оптимизации и отказоустойчивого проектирования. Приводится суперпозиционный метод синтеза куба функциональности для ее имплементации в кристаллы программируемой логики. Описываются аппаратно-ориентированные модели параллельного (за один цикл) вычисления булеана (множества всех подмножеств) на универсуме из n примитивов для решения задач покрытия, минимизации булевых функций, сжатия данных, синтеза и анализа цифровых систем за счет реализации процессорной структуры в форме диаграммы Хассе. Предлагается прототип квантового устройства, реализованного на основе программируемой логики, для оптимального решения задачи покрытия при анализе киберпространства.

1. Введение

Квантовые вычисления в последние годы становятся интересными для анализа кибернетического пространства, создания новых Интернет-технологий благодаря их некоторой альтернативности существующим моделям вычислительных процессов. Кроме того, рыночная привлекательность квантовых или кубитных моделей основывается на высоком параллелизме решения практически всех задач дискретной оптимизации, факторизации, минимизации булевых функций, эффективного сжатия, компактного представления и телепортации данных, отказоустойчивого проектирования [1-10] за счет существенного повышения аппаратных затрат. Но такая плата в настоящее время вполне допустима, поскольку существуют проблемы заполнения площадей силиконового кристалла, который содержит до 1 миллиарда вентилях при толщине пластины, равной 5 микрон. При этом современные технологии допускают создание пакета или «сэндвича», содержащего до 7 кристаллов, что соизмеримо с объемом нейронов головного мозга человека. Практически «беспроводное» соединение таких пластин основывается на технологической возможности сверления порядка 10 тысяч сквозных отверстий (vias) на 1 квадратном сантиметре. Наполнить полезной функциональностью такой объем допустимых на кристалле вентилях в настоящее время проблематично. Поэтому можно и нужно использовать «жадные» к аппаратуре модели и методы для создания. Благодаря дискретности и многозначности алфавитов описания информационных процессов, свойство параллелизма, заложенное в квантовых вычислениях, является особенно востребованным при создании эффективных и интеллектуальных «движков» для киберпространства или Интернета [11]; средств синтеза отказоустойчивых цифровых примитивов и систем [12]; тестирования и моделирования цифровых систем на кристаллах [13-15]; технологий защиты информации и компьютерных систем [5-7]. Здесь не рассматривается физическая основа квантовых вычислений, изначально заложенная в трудах ученых, ориентированных на возможность использования недетерминированных квантовых взаимодействий на уровне атома [16-17].

2. Кубитные, квантовые модели данных и вычислительных процессов

Квантовый компьютер предназначен для отказоустойчивого проектирования и решения оптимизационных задач, связанных с полным перебором на основе использования теории множеств. Особенность в том, что множество элементов в компьютере все равно упорядоченно, поскольку каждый байт имеет свой адрес. Поэтому теоретико-множественные операции сводятся к перебору всех адресов примитивных элементов. Адресный порядок структур данных хорош для задач, где компоненты моделей можно строго ранжировать, что дает возможность выполнять их анализ за один проход или одну итерацию. Там, где нет порядка в структуре, например, множество всех подмножеств, классическая модель памяти и вычислительных процессов наносит вред времени анализа ассоциации равных по рангу примитивов или, в лучшем случае, обработка ассоциативных групп является неэффективной. Что можно предложить для неупорядоченных данных вместо строгого порядка?

Процессор, где элементарной ячейкой служит образ или шаблон универсума из n примитивов, который генерирует $Q = 2^n$ всех возможных состояний такой ячейки в виде булеана или множества всех подмножеств. Прямое решение создания такой ячейки основано на унитарном позиционном кодировании состояний примитивов, которое с помощью суперпозиции последних образует множество всех подмножеств, формирующее в пределе универсум примитивов. Например, четыре примитива создают булеан, содержащий шестнадцать состояний (сочетаний), с помощью четырех двоичных разрядов:

$A=\{Q\}=(1000)$, $E=\{0100\}$, $H=\{0010\}$, $J=\{0001\}$, $O=\{Q,H\}=(1010)$, $I=\{E,J\}=(0101)$, $A=\{Q,E\}=(1100)$, $B=\{H,J\}=(0011)$, $S=\{Q,J\}=(1001)$, $P=\{E,H\}=(0110)$, $C=\{E,H,J\}=(1110)$, $F=\{Q,H,J\}=(1011)$, $L=\{Q,E,J\}=(1101)$, $V=\{Q,E,H\}=(1110)$, $Y=\{Q,E,H,J\}=(1111)$, $U=\{0000\}$.

Другая интерпретация булеана из четырех примитивов (двоичные коды: 00, 01, 10, 11), представленная ниже:

00	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
01	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1
10	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1
11	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0

создает 16 различных функций от двух переменных. В то же время последнюю таблицу можно представить как коды символов многозначного алфавита, которыми легко оперировать для решения задач синтеза и анализа булевых функций:

Q	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1
E	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1
H	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1
J	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
	∅	J	H	B	E	I	P	C	Q	S	O	F	A	L	V

Такую таблицу легко построить для любого числа примитивов ($n=2, 3, 4, 5, 6, 7, 8 \dots$), где теоретико-множественные операции над символами сводятся к логическим операциям над векторами.

Кубит есть двоичный вектор, содержащий n битов, для задания булеана (множества всех подмножеств) состояний $Q = 2^n$ на основе использования n примитивных символов (элементов).

Кубит – совокупность равнозначных двоичных n битов, формирующих единичным значением n примитивов, для обозначения $Q = 2^n$ состояний, составляющих булеан – множество всех подмножеств от n примитивов.

Здесь нет чисел! Все биты кубита равны при создании примитивов и отличаются только адресом. Любая теоретико-множественная операция выполняется за один такт, что невозможно при задании ассоциации примитивов на счетном (упорядоченном) пространстве памяти компьютера. Метрика (векторная и скалярная) анализа расстояний, предложенная в [11,13], абсолютно пригодная для измерения взаимодействия многозначных (двоичных) кубитных объектов, процессов и явлений путем использования хог-операции.

В идеале применение кубитной структуры дает возможность представить любую функциональность в виде двух кубов, привязанных к нулю и единице. Такие кубы формируют КНФ и ДНФ соответственно. Можно упрощать и далее путем исключения из рассмотрения нуля и единицы, неявно имея их в виду. При этом два куба, формирующие входные условия, будут всегда взаимно инверсными, поскольку они дополняют друг друга до универсума примитивов. Следовательно, необходимо оставлять лишь одну букву (символ), а значит один двоичный код, который есть таблица истинности функционального (двухвходового) примитива:

$$\begin{array}{|c|c|} \hline 00 & 0 \\ \hline 01 & 1 \\ \hline 10 & 1 \\ \hline 11 & 0 \\ \hline \end{array} = \begin{array}{|c|c|} \hline Q & 0 \\ \hline E & 1 \\ \hline H & 1 \\ \hline J & 0 \\ \hline \end{array} = \begin{array}{|c|c|} \hline S & 0 \\ \hline P & 1 \\ \hline \end{array} = \begin{array}{|c|c|} \hline 1001 & 0 \\ \hline 0110 & 1 \\ \hline \end{array} \rightarrow P = \boxed{0110}$$

$$Y = P = E \vee H = A_1 \vee A_2 = \bar{x}_1 x_2 \vee x_1 \bar{x}_2.$$

3. Кубит-процессор (Квантовый процессор)

Цель создания кубит-процессора – существенное уменьшение времени при решении задач оптимизации путем параллельного вычисления векторных логических операций над множеством всех подмножеств от примитивных компонентов за счет увеличения памяти для хранения промежуточных данных.

Задачи: 1) Определение структур данных для взятия булеана при решении задачи покрытия столбцов матрицы $M = [M_{ij}]$, $i = \overline{1, m}$; $j = \overline{1, n}$ единичными значениями строк. В частности, при $m = n = 8$ необходимо выполнить параллельно-логическую операцию над 256 вариантами всех возможных сочетаний векторов (строк матрицы), составляющих булеан. 2) Система команд процессора должна включать операции (and, or, xor) над векторами (словами) размерности m . 3) Разработка архитектуры кубит-процессора для параллельного вычисления $2^n - 1$ вариантов сочетаний, направленных на оптимальное решение NP-полной задачи покрытия. 4) Реализация прототипа кубит-процессора на базе программируемой логики PLD и верификация (валидация) аппаратного решения на примерах минимизации булевых функций. 5) Приведение других практических задач дискретной оптимизации к форме задачи покрытия для последующего решения на кубит-процессоре.

В качестве примера предлагается решить задачи поиска оптимального единичного покрытия всех столбцов минимальным числом строк матрицы M , представленной ниже:

M	1	2	3	4	5	6	7	8
a	1	.	.	.	.	1	.	.
b	.	.	1	.	.	.	1	.
c	1	.	.	.	1	.	1	.
d	.	1	.	1	.	.	.	1
e	.	1	.	.	1	.	.	.
f	1	.	1	.	.	1	1	.
g	.	1	.	1	.	.	.	1
h	.	.	1	.	1	.	.	.

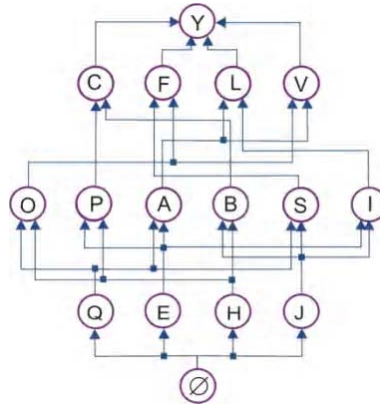
Для этого необходимо сделать перебор всех 255 сочетаний: из восьми по одной строке, по двум, трем, четверем, пяти, шести, семи и восьми. Минимальное количество примитивов (строк), формирующее покрытие, есть оптимальное решение. Таких решений может быть несколько. Диаграмма Хассе есть компромиссное предложение, относительно времени и памяти, или такая стратегия решения задачи покрытия, когда ранее полученный результат впоследствии используется для создания более сложной суперпозиции. Поэтому для каждой таблицы покрытия, содержащей n примитивов (строк), необходимо генерировать собственную мультипроцессорную структуру в форме диаграммы Хассе, которая далее должна быть использована для почти-параллельного решения NP-полной задачи. Например, для четырех строк таблицы покрытия диаграмма Хассе – структура мультипроцессора – будет иметь вид, представленный на рисунке.

Оптимальные решения задачи покрытия для матрицы M , которая генерирует 255 вариантов возможных сочетаний, представлены строками в форме ДНФ: $C = fgh \vee efg \vee cdf$.

Автомат управления вычислительным процессом для квантовой структуры путем восходящего анализа вершин графа основан на последовательном выполнении следующих шагов:

1. Занесение информации о примитивах в регистры (матрицы) $L_i^1 = P_i$ первого уровня с последующим анализом качества покрытия каждым примитивом в двоичном формате (1 – есть покрытие, 0 – нет покрытия). Если один из примитивов организует покрытие $\bigwedge_{j=1}^m L_{ij}^1 = 1$, процесс анализа структуры Хассе заканчивается. Иначе – выполнение перехода ($r = r + 1$)

на следующий уровень графа: $L_i^1 = P_i \rightarrow \bigwedge_{j=1}^m L_{ij}^1 = \begin{cases} 0 \rightarrow n = n + 1; \\ 1 \rightarrow \text{end.} \end{cases}$



Квантовая структура вычислительных процессов

2. Инициирование команды обработки очередного (второго) уровня. Последовательное

выполнение векторных (матричных) операций (or $L_i^r = L_{ij}^{r-1} \bigwedge_{j=1}^m L_{tj}^{r-1}$, and $\bigwedge_{j=1}^m L_{ij}^r = 1$) в целях анализа покрывающей способности сочетаний примитивных элементов r -уровня. Здесь $t = \overline{1, m}$, $i = \overline{1, m}$, $r = \overline{1, n}$, n – число ярусов или количество строк в таблице покрытия, m – число столбцов в ней. Если существует сочетание на рассматриваемом уровне, создающее полное покрытие, которое формирует оценку, равную 1, то обработка всех последующих уровней процессора не выполняется. В противном случае выполняется переход для анализа следующего уровня процессорной структуры:

$$L_i^r = L_{ij}^{r-1} \bigwedge_{j=1}^m L_{tj}^{r-1} \rightarrow \bigwedge_{j=1}^m L_{ij}^r = \begin{cases} 0 \rightarrow r = r + 1; \\ 1 \rightarrow \text{end.} \end{cases}$$

Достоинства кубитного Хассе процессора (Qubit Hasse Processor – QHP) заключаются в возможности использовать не более, чем двухвходовые схемы векторных логических операций (and, or, xor), а значит, в существенном уменьшении стоимости по Квайну реализации процессорных элементов (вершин) и памяти за счет применения последовательных вычислений и незначительного увеличения времени обработки всех вершин графа Хассе. Для каждой вершины используется критерий качества покрытия – наличие всех единиц в координатах вектора-результата. Если критерий качества выполняется, то все остальные вычисления можно не производить, поскольку диаграмма Хассе есть строго иерархическая структура по числу сочетаний в каждом ярусе. Это означает, что самое лучшее решение находится на более низком уровне иерархии. Варианты одного уровня равнозначны по реализации (стоимости), поэтому полученное первое качественное покрытие

($Q = \sum_{i=1}^n q_i = n$) есть лучшее решение, предполагающее остановку всех последующих

вычислений по стратегии диаграммы Хассе. Учитывая последовательно-параллельную стратегию анализа вершин графа, время (цикл) обработки всех примитивов QH-процессора определяется числом уровней иерархии (количеством битов (примитивов, строк в таблице покрытия) в кубитной переменной), умноженным на время анализа одной вершины:

$T = \log_2 2^n \times t = t \times n$. При этом длина m строки таблицы покрытия не влияет на оценку быстродействия. Анализ вершины включает две команды: логическую (and, or, xor) и операцию вычисления критерия качества покрытия в форме скаляра путем применения функции and ко всем разрядам вектора-результата:

$$m_{ir,j} = M_{i,j} \vee M_{r,j}, (j = \overline{1, n}; \{i \neq r\} = \overline{1, m});$$

$$m_{ir}^S = \bigwedge m_{ir,j} = \bigwedge (M_{i,j} \vee M_{r,j})$$

Аппаратные затраты на реализацию QН-процессора зависят от суммарного числа вершин графа Хассе и от количества битов (разрядов) в строке таблицы покрытия:

$$H = 2^n \times k \times m,$$

где k – коэффициент аппаратной реализации (сложности) одного разряда бинарной векторной логической операции и последующей команды вычисления критерия качества покрытия.

Таким образом, высокое быстродействие решения задачи покрытия достигается существенным повышением аппаратных затрат (в $2^n \times k \times m / k \times m \times n = 2^n / n$ раз по сравнению с последовательной обработкой графовых вершин), которое обеспечивает компромиссный вариант между полностью параллельной структурой вычислительных процессов (здесь затраты аппаратуры определяются числом примитивов в каждой вершине

$H = k \times m \times n \times 2^n$, а увеличение аппаратуры составляет 2^n раз) и последовательными вычислениями однопроцессорного компьютера (здесь быстродействие обработки графа

Хассе равно $T = t \times 2^n$, а аппаратные затраты равны $H^* = k \times m \times n$). Уменьшение аппаратуры по сравнению с параллельным вариантом обработки графа составляет

$Q^H = k \times m \times n \times 2^n / k \times m \times 2^n = n$. Как следствие существенной аппаратной избыточности, уменьшение времени анализа вершин графа по сравнению с последовательной обработкой структуры имеет следующую оценку:

$$Q^T = \frac{t \times 2^n}{t \times n} = \frac{2^n}{n}.$$

Реализация вычислительного устройства выполнена на основе кристалла FPGA фирмы Xilinx xc3s1600e-4-fg484, где основные параметры имеют следующий вид:

Листинг Map-report

Logic Utilization:

Number of Slice Flip Flops: 2,286 out of 29,504 7%

Number of 4 input LUTs: 2,715 out of 29,504 9%

Logic Distribution:

Number of occupied Slices: 1,514 out of 14,752 10%

Number of Slices containing only related logic: 1,514 out of 1,514 100%

Number of Slices containing unrelated logic: 0 out of 1,514 0%

*See NOTES below for an explanation of the effects of unrelated logic.

Total Number of 4 input LUTs: 2,715 out of 29,504 9%

Number of bonded IOBs: 321 out of 376 85%

Number of BUFGMUXs: 1 out of 24 4%

Timing parameters of project:

Tclk_to_clk = 4.672 ns

Tclk_to_pad_max = 11.552 ns

Period = max{ Tclk_to_clk, Tclk_to_pad_max };

Period = 11.552 ns

Fclk = 86,5 МГц

4. Заключение

Реализация квантового процессора на основе структуры Хассе позволила в n раз уменьшить аппаратные затраты по сравнению с параллельной реализацией, что в свою очередь уменьшило быстродействие процессора также в n раз. Вывод – необходимо создавать новые структуры данных для снижения аппаратной стоимости квантовых вычислений и/или более интеллектуальные алгоритмы решения задачи покрытия на диаграммах Хассе. Синтез модели функциональности, представленной в форме одного куба, можно использовать при создании PLD-ориентированных прототипов вычислительных устройств, когда знание куба функциональности приводит к тривиальной технологии имплементации ее в кристалл.

Научная новизна – впервые предложена модель данных и структура аппаратной реализации квантового компьютера, которая характеризуется использованием структуры Хассе, что дает возможность существенно (x100) повысить быстродействие решения практических задач дискретной оптимизации. Предложен суперпозиционный метод синтеза куба функциональности, который характеризуется использованием структуры примитивных элементов и для определенного класса устройств позволяет существенно уменьшить время получения модели цифрового устройства, ориентированной на имплементацию в кристаллы PLD.

Список литературы: 1. *Beth T.* Quantum computing: an introduction // Proceedings of the IEEE International Symposium on Circuits and Systems, ISCAS. 2000. Geneva. Vol. 1. P. 735 – 736. 2. *Jonker P., Jie Han.* On quantum and classical computing with arrays of superconducting persistent current qubits // Proceedings of Fifth IEEE International Workshop on Computer Architectures for Machine Perception. 2000. P. 69 – 78. 3. *Keyes R. W.* Challenges for quantum computing with solid-state devices // Computer. Jan. 2005. Vol. 38, Issue 1. P. 65 – 69. 4. *Glassner A.* Quantum computing. 3. // IEEE Computer Graphics and Applications. Nov/Dec 2001. Vol. 21, Issue 6. P. 72 – 82. 5. *Marinescu D. C.* The Promise of Quantum Computing and Quantum Information Theory – Quantum Parallelism // Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium (IPDPS'05). 2005. P. 112-114. 6. *Oskin M., Chong F. T., Chuang I. L.* A practical architecture for reliable quantum computers // Computer. Jan. 2002. Vol. 35, No. 1. P. 79-87. 7. *A. Glassner.* Quantum Computing, Part 1 // IEEE Computer Graphics and Applications. – July/August 2001. P. 84-92. 8. *P. Aliferis, F. Brito, D. P. DiVincenzo, J. Preskill, M. Steffen, B. M. Terhal.* Fault-tolerant computing with biased-noise superconducting qubits // New Journal of Physics. January 30, 2009. 19 p. http://iopscience.iop.org/1367-2630/11/1/013061/pdf/1367-2630_11_1_013061.pdf. 9. *Hunting B., Mertz D.* Introduction to Quantum Computing // <http://www.ibm.com/developerworks/linux/library/l-quant/index.html>. 10. *DiVincenzo D. P.* The Physical Implementation of Quantum Computation // IBM T.J. Watson Research Center, Yorktown Heights, NY 10598 USA. July 9, 2004. Preprint available online http://arxiv.org/PS_cache/quant-ph/pdf/0002/0002077v3.pdf. 11. *Инфраструктура мозгоподобных вычислительных процессов / М.Ф. Бондаренко, О.А. Гузь, В.И. Хаханов, Ю.П. Шабанов-Кушнарченко.* Харьков: Новое слово. 2010. 160 с. 12. *Mark Gregory Whitney.* Practical Fault Tolerance for Quantum Circuits. PhD Dissertation in Computer Science. Berkeley: University of California. 2009. 206p. 13. *Хаханов В. И., Литвинова Е. И., Чумаченко С. В., Гузь О.А.* Логический ассоциативный вычислитель. Электронное моделирование. 2011. № 1. С. 73-90. 14. *Hahanov V., Wajeb Gharibi, Litvinova E., Chumachenko S.* Information analysis infrastructure for diagnosis. Information. An international interdisciplinary journal. 2011. Japan. Vol. 14., No 7. P. 2419-2433. 15. *Хаханов В.И.* Проектирование и тестирование цифровых систем на кристаллах / В.И. Хаханов, Е.И. Литвинова, О.А. Гузь. Харьков: ХНУРЭ, 2009. 484с. 16. *Stig Stenholm, Kalle-Antti Suominen.* Quantum approach to informatics. Published by John Wiley & Sons, Inc., Hoboken, New Jersey. 2005. 238p. 17. *Michael A. Nielsen & Isaac L. Chuang.* Quantum Computation and Quantum Information. Published in the United States of America by Cambridge University Press, New York. 2010. 676p.

Поступила в редколлегию 11.10.2012

Хаханова Ирина Витальевна, д-р техн. наук, профессор кафедры автоматизации проектирования вычислительной техники ХНУРЭ. Научные интересы: проектирование и тестирование цифровых систем и сетей на кристаллах. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-326. E-mail: hahanov@kture.kharkov.ua.

МОДЕЛИ УПРАВЛЕНИЯ УЯЗВИМОСТЬЮ

Предлагается математический аппарат создания инфраструктуры программно-аппаратных телекоммуникационных информационных кибернетических систем (КС), ориентированной на защиту от несанкционированного доступа к сервисам, определенным в спецификации системы, путем проникновения через легальные интерфейсы взаимодействия компонентов, имеющие уязвимости. Инфраструктура защитных сервисов создается вместе с киберсистемой и сопровождает последнюю в течение всего жизненного цикла, обслуживая все последующие модификации КС, и сама постоянно повышает свой интеллект путем пополнения истории и библиотек конструктивных и деструктивных компонентов.

1. Модель процессов тестирования уязвимостей и проникновений

Предлагаются технологичные и эффективные процесс-модели и методы диагностирования уязвимостей или функциональных нарушений в программных и/или аппаратных компонентах киберсистемы. Используются регистровые (векторные) или матричные (табличные) структуры данных, которые ориентированы на параллельное выполнение логических операций при поиске уязвимых компонентов КС.

Проблема синтеза или анализа компонентов произвольной системы и инфраструктуры сервисов может быть сформулирована в виде взаимодействия (симметрической разности – аналог хог-операции на булеане) в кибернетическом пространстве ее модели F с входными воздействиями T и реакциями L : $f(F, T, L) = \emptyset \rightarrow F \Delta T \Delta L = \emptyset$.

Киберпространство – совокупность взаимодействующих по метрике информационных процессов и явлений, использующих в качестве носителя компьютерные системы и сети. В частности, компонент пространства представлен k -мерным (кортежем) вектором $a = (a_1, a_2, \dots, a_j, \dots, a_k)$, $a_j = \{0, 1\}$ в двоичном алфавите. Нуль-вектор есть k -мерный кортеж, все координаты которого равны нулю: $a_j = 0, j = \overline{1, k}$.

Метрика β кибернетического (двоичного) пространства определяется единственным равенством, которое формирует нуль-вектор для хог-суммы расстояний d_i между ненулевым и конечным числом точек (объектов), замкнутых в цикл: $\beta = \bigoplus_{i=1}^n d_i = 0$.

Иначе: метрика β векторного логического двоичного пространства есть равная нулю-вектору хог-сумма расстояний между конечным числом точек (вершин) графа, образующих цикл. Сумма n -мерных двоичных векторов, задающих координаты точек цикла, равна нуль-вектору. Данное определение метрики оперирует отношениями, что позволяет сократить систему аксиом с трех до одной и распространить ее действие на любые конструкции n -мерного киберпространства.

Метрика β кибернетического многозначного пространства, где каждая координата вектора (объекта) определена в алфавите, составляющем булеан на универсуме примитивов мощностью p : $a_j = \{\alpha_1, \alpha_2, \dots, \alpha_r, \dots, \alpha_m\}$, $m = 2^p$, есть равная $\emptyset$ -вектору (по всем координатам) симметрическая разность расстояний между конечным числом точек, обра-

зующих цикл: $\beta = \bigtriangleup_{i=1}^n d_i = \emptyset$.

Равенство пустому вектору симметрической разности по координатному теоретико-множественного взаимодействия подчеркивает равнозначность компонентов (расстояний), формирующих уравнения.

Введенная метрика представляет не только теоретический интерес, но имеет и практическую направленность на обобщение и классификацию задач тестирования путем создания модели хог-отношений на множестве из четырех основных компонентов. Процедуры синтеза тестов, моделирования уязвимостей и их диагностирования можно свести к хог-отношениям на графе (рис. 1) полного взаимодействия четырех вершин (функциональность, киберсистема (Unit Under Test), тест, уязвимости) $G = \{F, U, T, L\}$.

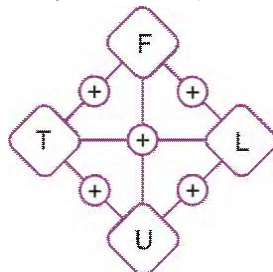


Рис. 1. Граф взаимодействия компонентов тестирования

Такой граф порождает четыре базовых треугольника, которые формируют 12 практически полезных триад отношений, формулирующих задачи тестирования:

$T \oplus F \oplus L = 0$	$T \oplus L \oplus U = 0$	$T \oplus F \oplus U = 0$	$F \oplus L \oplus U = 0$
1) $T = F \oplus L$	4) $T = L \oplus U$	7) $T = F \oplus U$	10) $F = L \oplus U$
2) $F = T \oplus L$	5) $L = T \oplus U$	8) $F = T \oplus U$	11) $L = F \oplus U$
3) $L = T \oplus F$	6) $U = T \oplus L$	9) $U = T \oplus F$	12) $U = F \oplus L$

Введение вершины U в граф взаимодействия компонентов тестирования расширяет функциональные возможности модели, появляются новые свойства полученной системы. Введение в структуру новой вершины должно иметь весомые аргументы в пользу ее целесообразности. Что касается представленного на рис. 1 графа, содержательно все формульно описанные задачи можно классифицировать в группы следующим образом.

Группа 1 – теоретические эксперименты (на модели функциональности), без киберсистемы: 1) Синтез теста по модели функциональности для заданного списка уязвимостей. 2) Построение модели функциональности на основе заданного теста и списка уязвимостей. 3) Моделирование уязвимостей функциональности на заданном тесте.

Группа 2 – реальные эксперименты (на киберсистеме), без модели функциональности: 4) Синтез теста путем физической эмуляции уязвимостей в КС. 5) Определение списка уязвимостей устройства при выполнении диагностического эксперимента. 6) Верификация теста и уязвимостей в эксперименте на реальной КС.

Группа 3 – тестовые эксперименты (верификация), без уязвимостей: 7) Синтез теста путем сравнения результатов моделирования модели и реальной КС. 8) Синтез функциональности по реальной КС и заданному тесту. 9) Верификация теста и модели функциональности относительно реальной КС с существующими уязвимостями.

Группа 4 – эксперименты в процессе функционирования, на рабочих воздействиях: 10) Проверка правильности поведения реальной КС на существующих или заданных уязвимостях. 11) Проверка работоспособности устройства относительно существующей модели в процессе функционирования. 12) Верификация функциональности и списка уязвимостей относительно поведения реального КС.

Наиболее популярными задачами из перечисленного выше списка являются: 1, 3, 5, 8, 9. Можно ввести и другую классификацию типов задач, которая дает возможность определить на графе $G = (F, U, T, L)$ все концептуальные пути решения целевых проблем: синтеза тестов, определения модели функциональности, генерирования модели уязвимостей и проектирования КС:

- 1) $T = F \oplus L$; 4) $F = T \oplus L$; 7) $L = T \oplus F$; 10) $U = T \oplus L$;
- 2) $T = U \oplus L$; 5) $F = U \oplus L$; 8) $L = T \oplus U$; 11) $U = T \oplus F$;
- 3) $T = F \oplus U$; 6) $F = T \oplus U$; 9) $L = F \oplus U$; 12) $U = F \oplus L$.

Все конструкции, представленные в отношениях, обладают замечательным свойством обратимости. Компонент, вычисляемый с помощью двух других, может быть использован в качестве аргумента для определения любого из двух исходных. Потому здесь можно говорить о транзитивной обратимости каждой триады отношений на полном графе, когда по двум любым компонентам всегда и однозначно можно восстановить или определить третий. При этом формат представления каждого компонента должен быть одинаковым по форме и размерности (векторы, матрицы). На основе предложенной метрики и моделей тестирования далее рассмотрим более подробно методы диагностирования уязвимостей или функциональных нарушений.

2. Граф-метод поиска функциональных нарушений в КС

Используется уравнение пространства

$$f(F, T, L, U) = 0 \rightarrow F \oplus T \oplus L \oplus U = 0,$$

которое трансформируется к виду $L = (T \oplus F) \oplus (T \oplus U)$. Диагностирование уязвимостей (функциональных нарушений) сводится к сравнению результатов модельного $(T \oplus F)$ и натурального $(T \oplus U)$ экспериментов, которое формирует список функциональных нарушений L , присутствующих в объекте диагностирования. Формула-модель процесса поиска блока F_i с функциональными нарушениями сводится к выбору решения посредством определения хог-взаимодействия между тремя компонентами:

$$L = F_i \leftarrow \bigoplus_{i=1}^p (T \oplus F_i) \oplus (T \oplus U_i) = 0.$$

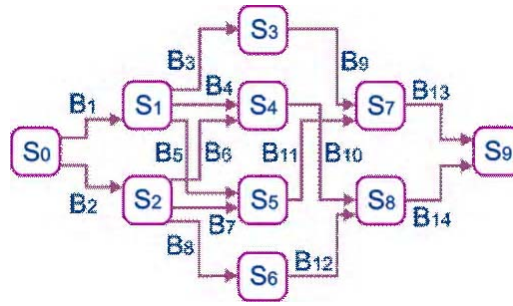
Аналитическая модель верификации HDL-кода с использованием механизма ассерций (дополнительных линий наблюдения) ориентирована на достижение заданной глубины диагностирования и представлена в следующем виде:

$$\begin{aligned} M &= f(F, A, B, S, T, L), & F &= (A * B) \times S; S = f(T, B); \\ A &= \{A_1, A_2, \dots, A_i, \dots, A_n\}; & B &= \{B_1, B_2, \dots, B_i, \dots, B_n\}; \\ S &= \{S_1, S_2, \dots, S_i, \dots, S_m\}; & S_i &= \{S_{i1}, S_{i2}, \dots, S_{ij}, \dots, S_{ip}\}; \\ T &= \{T_1, T_2, \dots, T_i, \dots, T_k\}; & L &= \{L_1, L_2, \dots, L_i, \dots, L_n\}. \end{aligned}$$

Здесь $F = (A * B) \times S$ – функциональность, представленная графом (рис. 2) транзакций программных блоков (Code-Flow Transaction Graph – CFTG), где $S = \{S_1, S_2, \dots, S_i, \dots, S_m\}$ – вершины или состояния программного продукта при моделировании тестовых сегментов. Иначе граф можно идентифицировать как ABC-граф – Assertion Based Coverage Graph. Каждое состояние $S_i = \{S_{i1}, S_{i2}, \dots, S_{ij}, \dots, S_{ip}\}$ определяется значениями существенных переменных КС (переменные, память, терминалы, компьютеры). Ориентированные дуги графа представлены совокупностью функциональных блоков:

$$B = (B_1, B_2, \dots, B_i, \dots, B_n), \quad \bigcup_{i=1}^n B_i = B; \quad \bigcap_{i=1}^n B_i = \emptyset,$$

где каждому из них может быть поставлена в соответствие ассерция $A_i \in A = \{A_1, A_2, \dots, A_i, \dots, A_n\}$. Каждая дуга B_i – группа операторов кода – формирует состояние вершины $S_i = f(T, B_i)$ в зависимости от теста $T = \{T_1, T_2, \dots, T_i, \dots, T_k\}$. Каждой вершине может быть поставлен ассерционный монитор, объединяющий ассерции входящих в вершину дуг $A(S_i) = A_{i1} \vee A_{i2} \vee \dots \vee A_{ij} \vee \dots \vee A_{in}$. Вершина может иметь более одной входящей (исходящей) дуги. Множество блоков с функциональными нарушениями представлено списком $L = \{L_1, L_2, \dots, L_i, \dots, L_n\}$.



$$\begin{aligned}
 B &= (B_1 B_3 B_9 \vee (B_2 B_7 \vee B_1 B_5) B_{11}) B_{13} \vee \\
 &\vee ((B_1 B_4 \vee B_2 B_6) B_{10} \vee B_2 B_8 B_{12}) B_{14} = \\
 &= B_1 B_3 B_9 B_{13} \vee B_2 B_7 B_{11} B_{13} \vee B_1 B_5 B_{11} B_{13} \vee \\
 &\vee B_1 B_4 B_{10} B_{14} \vee B_2 B_6 B_{10} B_{14} \vee B_2 B_8 B_{12} B_{14}.
 \end{aligned}$$

Рис. 2. Пример ABC-графа для HDL-кода

Модель HDL-кода, представленная в форме ABC-графа, отображает не только структуру программного кода, но и тестовые срезы функциональных покрытий, формируемые с помощью программных блоков, входящих в рассматриваемую вершину. Последняя определяет отношение между достигнутым на тесте пространством переменных и потенциально возможным, которое формирует функциональное покрытие как мощность состояния i -

вершины графа $Q = \text{card}C_i^r / \text{card}C_i^p$. В совокупности все вершины графа должны составлять полное покрытие пространства состояний переменных КС, которое формирует качество теста, равное 1 (100%): $Q = \text{card} \bigcup_{i=1}^m C_i^r / \text{card} \bigcup_{i=1}^m C_i^p = 1$. Кроме того, механизм ассерций

$\langle A, C \rangle$, существующий на графе, позволяет выполнять мониторинг дуг (function-coverage) $A = \{A_1, A_2, \dots, A_i, \dots, A_n\}$ и вершин (state-coverage) $C = \{C_1, C_2, \dots, C_i, \dots, C_m\}$. Ассерции на дугах отвечают за диагностирование функциональных нарушений в КС блоках. Ассерции на вершинах графа дают дополнительную информацию о качестве тестов (ассерций) в целях их улучшения или дополнения. Транзакционный граф КС блоков дает возможность: 1) использовать аппарат тестопригодного проектирования для оценки качества КС; 2) оценить затраты на создание тестов, диагностирование и исправление функциональных нарушений; 3) оптимизировать синтез теста путем решения задачи покрытия минимальным множеством активизированных путей всех дуг (вершин). Например, минимальный тест для приведенного ABC-графа имеет шесть сегментов, которые активизируют все существующие пути:

$$\begin{aligned}
 T &= S_0 S_1 S_3 S_7 S_9 \vee S_0 S_1 S_4 S_8 S_9 \vee S_0 S_1 S_5 S_7 S_9 \vee \\
 &\vee S_0 S_2 S_4 S_8 S_9 \vee S_0 S_2 S_5 S_7 S_9 \vee S_0 S_2 S_6 S_8 S_9.
 \end{aligned}$$

Тесту можно поставить в соответствие следующую матрицу активизации программных блоков:

B_{ij}	B_1	B_2	B_3	B_4	B_5	B_6	B_7	B_8	B_9	B_{10}	B_{11}	B_{12}	B_{13}	B_{14}
T_1	1	.	1	.	.	.	.	.	1	.	.	.	1	.
T_2	1	.	.	1	.	.	.	.	.	1	.	.	.	1
T_3	1	.	.	.	1	.	.	.	.	.	1	.	1	.
T_4	.	1	.	.	.	1	.	.	.	1	.	.	.	1
T_5	.	1	.	.	.	.	1	.	.	.	1	.	1	.
T_6	.	1	.	.	.	.	.	1	.	.	.	1	.	1

Матрица активизации иллюстрирует факт неразличимости на тесте функциональных нарушений в блоках 3 и 9, 8 и 12, которые составляют два класса эквивалентностей при

наличии одной ассерции (монитора) в вершине 9. Для устранения такой неразличимости необходимо создать два дополнительных монитора в вершинах 3 и 6. В результате 3 ассерции на вершинах $A = (A_3, A_6, A_9)$ дают возможность различить между собой все блоки программного кода. Таким образом, граф позволяет не только синтезировать оптимальный тест, но и определять минимальное число ассерционных мониторов здесь и далее в вершинах для поиска неисправных блоков с заданной глубиной диагностирования.

Увеличение числа ассерционных мониторов приводит к модификации таблицы активизации. Иначе, на заданном тесте и механизме ассерций необходимо однозначно решать задачу диагностирования функциональных нарушений КС с глубиной до функционального блока. При этом число ассерций и тестовых сегментов должно быть минимально допустимым для кодовой идентификации всех блоков:

$$|T| + |A| \geq \log_2 |B| = \text{card}T + \text{card}A \geq \log_2 \text{card}B.$$

Первоначально количество ассерционных мониторов равно числу тестовых сегментов. Таблица активизации функциональных модулей позволяет выполнять идентификацию блоков кода с функциональными нарушениями на обобщенном векторе экспериментальной проверки (ассерционного мониторинга):

$$V = (V_1, V_2, \dots, V_i, \dots, V_n), V_i = \{0, 1\}, V_i = T_i \oplus B_j, \forall j (B_{ij} = 1).$$

Координата вектора $V_i = T_i \oplus B_j = 1$ идентифицирует факт «падения» тест-сегмента на подмножестве активизированных им блоков. В соответствии с вектором V , заданным на таблице активизации с учетом приведенного выше правила вычисления его координат:

B_{ij}	B_1	B_2	B_3	B_4	B_5	B_6	B_7	B_8	B_9	B_{10}	B_{11}	B_{12}	B_{13}	B_{14}	V
T_1	1	.	1	.	.	.	.	.	1	.	.	.	1	.	0
T_2	1	.	.	1	.	.	.	.	.	1	.	.	.	1	1
T_3	1	.	.	.	1	.	.	.	.	.	1	.	1	.	0
T_4	.	1	.	.	.	1	.	.	.	1	.	.	.	1	1
T_5	.	1	.	.	.	.	1	.	.	.	1	.	1	.	0
T_6	.	1	.	.	.	.	.	1	.	.	.	1	.	1	1

строится логическая функция функциональных нарушений КС, которая упрощается на основе использования координат вектора экспериментальной проверки V :

$$\begin{aligned} B &= (\bar{T}_1 \vee B_1 \vee B_3 \vee B_9 \vee B_{13}) \wedge (\bar{T}_2 \vee B_1 \vee B_4 \vee B_{10} \vee B_{14}) \wedge \\ &\wedge (\bar{T}_3 \vee B_1 \vee B_5 \vee B_{11} \vee B_{13}) \wedge (\bar{T}_4 \vee B_2 \vee B_6 \vee B_{10} \vee B_{14}) \wedge \\ &\wedge (\bar{T}_5 \vee B_2 \vee B_7 \vee B_{11} \vee B_{13}) \wedge (\bar{T}_6 \vee B_2 \vee B_8 \vee B_{12} \vee B_{14}); \\ \{V, T\} &= (010101) \rightarrow \\ B &= (0 \vee B_1 \vee B_4 \vee B_{10} \vee B_{14}) \wedge (0 \vee B_2 \vee B_6 \vee B_{10} \vee B_{14}) \wedge \\ &\vee (0 \vee B_2 \vee B_8 \vee B_{12} \vee B_{14}) = \\ &= (B_1 \vee B_4 \vee B_{10} \vee B_{14}) \wedge (B_2 \vee B_6 \vee B_{10} \vee B_{14}) \wedge \\ &\vee (B_2 \vee B_8 \vee B_{12} \vee B_{14}) = \\ &= B_1 B_2 \vee B_4 B_2 \vee \dots \vee B_3 B_6 B_{12} \vee \dots \vee B_{14}. \end{aligned}$$

После преобразования конъюнктивной нормальной формы (КНФ) к дизъюнктивной нормальной форме полученные термы содержат все возможные решения в виде покрытия единичных координат вектора экспериментальной проверки одиночными или кратными функциональными нарушениями КС блоков. Выбор лучшего решения связан с определением терма ДНФ минимальной длины. В данном примере оптимальным решением является терм, содержащий один блок $B = B_{14}$, который своим функциональным нарушением покрывает три единицы в векторе экспериментальной проверки $V = (010101)$. Данный факт также очевиден из сравнения двух последних столбцов матрицы активизации B .

Другое аппаратно-ориентированное аналитическое решение связано с синтезом много-выходовой комбинационной схемы – дешифратора по матрице активизации программных блоков:

$$B_1 = T_1 T_2 T_3 \bar{T}_4 \bar{T}_5 \bar{T}_6; B_2 = \bar{T}_1 \bar{T}_2 \bar{T}_3 T_4 T_5 T_6;$$

$$B_3 = \bar{T}_1 T_2 T_3 T_4 T_5 T_6; \dots B_{14} = \bar{T}_1 T_2 \bar{T}_3 T_4 \bar{T}_5 T_6.$$

Такое устройство имеет число входов, равное количеству тестовых сегментов, а выходных линий – равное числу КС блоков. При подаче на входы дешифратора вектора экспериментальной проверки формируется единичное значение на одном из его выходов. При этом номер выхода соответствует блоку, имеющему функциональные нарушения. Такая схема представляет интерес для создания замкнутой в цикл инфраструктуры тестирования и исправления функциональных нарушений, где адрес уязвимого блока может быть использован для его автоматической замены на альтернативный модуль из существующей библиотеки диверсных решений.

3. Векторно-логические методы диагностирования уязвимостей

Основная цель методов – определение места причины и вида уязвимости на структуре КС путем анализа таблиц уязвимостей (функциональных нарушений – ФН), построенных в процессе моделирования всех возможных деструктивов. Аналитическая модель проверки КС с использованием механизма ассерций ориентирована на достижение заданной глубины диагностирования уязвимостей и представлена в следующем виде:

$$M = f(F, L, T, C, A, t); C = \{C_1, C_2, \dots, C_i, \dots, C_m\}; L = \{L_1, L_2, \dots, L_i, \dots, L_n\};$$

$$A(t) = \{A_1, A_2, \dots, A_i, \dots, A_k\}; A \subseteq L; F = L \times C; k \leq n; T = \{T_1, T_2, \dots, T_i, \dots, T_p\}.$$

Здесь C_i – группа операторов кода, нагруженная на вершину (компонент КС) L_i и формирующая ее состояние; F – функциональность, представленная транзакционным графом $F = L \times C$ в виде декартова произведения множества вершин и дуг; A – совокупность ассерций, как подмножество вершин транзакционного графа $A \subseteq L$. Метод поиска функциональных нарушений блока киберсистемы использует предварительно построенную таблицу ФН $B = [B_{ij}]$, где строка есть отношение между тестовым сегментом и подмножеством блоков $T_i \approx (B_{i1}, B_{i2}, \dots, B_{ij}, \dots, B_{in})$ с возможными ФН. Столбец таблицы формирует отношение между блоком и тестовыми сегментами $B_j \approx (T_{1j}, T_{2j}, \dots, T_{ij}, \dots, T_{pj})$, которые могут проверять блок с ФН. На стадии моделирования определяется обобщенная реакция $m = \{m_1, m_2, \dots, m_i, \dots, m_p\}$ механизма ассерций F на тест, путем формирования $m_i = (A_1 \vee A_2 \vee \dots \vee A_i \vee \dots \vee A_k)$, $A_i = \{0, 1\}$ как реакции ассерций на тест-сегмент T_i . Поиск ФН основан на определении хог-операции между вектором состояния ассерций и столбцов таблицы ФН $m \oplus (B_1 \vee B_2 \vee \dots \vee B_j \vee \dots \vee B_n)$. Выбор решения определяется совокупностью векторов B_j с минимальным числом единичных координат

$$B = \min_{j=1, n} [B_j = \sum_{i=1}^p (B_{ij} \oplus m_i)],$$

формирующих программные блоки с ФН, проверяемые на тестовых сегментах.

Повышение адекватности модели диагностирования ФН блоков КС связано с расширением пространства реакции проекта в процессе его тестовой верификации. В этом случае следует расширить пространство существования механизма ассерций до двумерного путем модификации векторов m и A :

$$A_i = (A_{i1}, A_{i2}, \dots, A_{ij}, \dots, A_{ik}); m_i = (m_{i1}, m_{i2}, \dots, m_{ij}, \dots, m_{ik});$$

$$B_{ij} = (B_{ij1}, B_{ij2}, \dots, B_{ijr}, \dots, B_{ijk}); \{A_{ij}, m_{ij}, B_{ijr}\} = \{0, 1\}; i = \overline{1, p}; j = \overline{1, n}; r = \overline{1, k}.$$

При этом таблица ФН становится трехмерной по параметрам: $i = \overline{1, p}$; $j = \overline{1, n}$; $r = \overline{1, k}$ – числа строк или тест-сегментов, столбцов или функциональных блоков, ассерций или точек наблюдения для блоков соответственно:

B	A ₁	A ₂	A ₃	A ₄	A ₅	A ₆	A ₇	A ₈	m
	B ₁	B ₂	B ₃	B ₄	B ₅	B ₆	B ₇	B ₈	
T ₁	1001	1110	0001	0110	0011	1110	0001	1110	1001
T ₂	1010	1000	0001	0110	0001	0001	1110	1110	1000
T ₃	1110	0001	1111	0111	1011	1110	1001	1101	0011
T ₄	1000	0111	0110	1001	1000	0001	1011	1110	1011
T ₅	0001	0110	1100	0111	0010	1110	0001	1000	1001
T ₆	1110	1011	0110	0001	1110	1100	1001	0111	1110
T ₇	0110	1101	0110	1000	0111	1011	0101	1110	0001
T ₈	1000	1110	1001	0100	0101	1001	0111	0001	1110

Потенциально трехмерность таблицы ФН увеличивает объем диагностической информации, что дает возможность повысить глубину поиска уязвимости, в пределах до неделимого примитива КС. Однако для анализа трехмерной таблицы ФН необходимо модифицировать предложенные методы диагностирования.

Поиск уязвимостей по таблице функциональных нарушений на основе вектора экспериментальной проверки – реакции (m) киберсистемы на тест $m = (m_1, m_2, \dots, m_i, \dots, m_n)$, $m_i = \{0, 1\}$ сводится к методам анализа строк или столбцов. Первый метод основан на применении векторной хог-операции между реакцией m КС на тест, формально рассматриваемый в качестве входного вектор-столбца или маски m, и столбцов таблицы ФН $m \oplus (B_1 \vee B_2 \vee \dots \vee B_j \vee \dots \vee B_m)$. Для подсчета качества взаимодействия векторов $Q_j(m \oplus B_j)$ в целях выбора лучшего решения определяются столбцы с минимальным числом единиц результирующего вектора:

$$B = \min_{j=1, k} [B_j = \sum_{i=1}^n (B_{ij} \oplus m_i)].$$

Они идентифицируют и формируют примитивы с функциональными нарушениями, проверяемые на тестовых наборах. Аналитическая модель процесса получения решения в виде списка блоков с уязвимостями, присутствующих в КС, представлена в следующем виде:

$$B = \min_{j=1, k} [B_j = \sum_{i=1}^n (B_{ij} \oplus m_i)]; B^0 = \bigvee_{j=1, k} [B_j = \sum_{i=1}^n (B_{ij} \oplus m_i) = 0].$$

Здесь фигурирует вектор экспериментальной проверки, который является входным для последующего анализа таблицы ФН $m = f(A, B) \oplus f^*(A, B, L)$ есть результат проведения тестового эксперимента – сравнение функционалов (состояний выходов) эталонной $f(A, B)$ и реальной $f^*(A, B, L)$ КС с ФН L на тестовых наборах A. Во втором случае, если множество $B^0 > 1$, это означает наличие эквивалентных, не различимых на данном тесте и механизме ассерций, функциональных нарушений. Выбор лучшего решения $Y = \vee[(Q_1 \wedge Q_2) \oplus Q_1]$ с минимальным числом единичных координат из двух альтернатив, представленных векторными оценками $Q_1 \vee Q_2$, осуществляется с помощью структуры, изображенной на рис. 3.

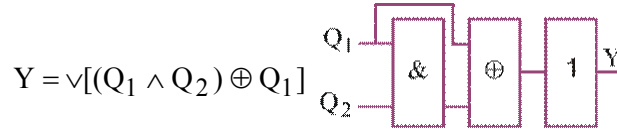


Рис. 3. Схема выбора лучшего решения

Достоинство метода – выбор лучшего решения из всех возможных одиночных и кратных ФН. По существу, в список ФН включаются такие одиночные ФН, которые при логическом умножении на вектор экспериментальной проверки дают результат в виде соответствующего вектор-столбца. Дизъюнкция всех столбцов, составляющих решение,

равна вектору экспериментальной проверки $\bigvee_{j=1}^r (B_j \in B) = m$.

Вычислительная сложность метода анализа столбцов определяется следующей зависимостью: $Z^c = 3n^2 + n^2 = 4n^2$; $Z^r = 3n + n = 4n$. Здесь первая оценка учитывает выполнение координатных операций над матрицей, размерностью $n \times n$. Вторая оценка определяет вычислительную сложность регистровых параллельных операций для подсчета критериев качества и обработки матрицы соответственно.

Логический метод анализа строк таблицы ФН. Стратегия определения ошибок программного кода по таблице ФН связана с анализом ее строк, состоящим из двух процедур: 1) вычисление логического произведения конъюнкции строк, отмеченных единичными значениями вектора $T_i (m_i = 1)$, на отрицание дизъюнкции нулевых строк $T_i (m_i = 0)$ для одиночных дефектных блоков; 2) вычисление логического произведения дизъюнкции единичных строк на отрицание дизъюнкции нулевых строк для кратных дефектных блоков:

$$B^s = \left(\bigwedge_{\forall m_i=1} T_i \right) \wedge \left(\overline{\bigvee_{\forall m_i=0} T_i} \right); B^m = \left(\bigvee_{\forall m_i=1} T_i \right) \wedge \left(\overline{\bigvee_{\forall m_i=0} T_i} \right); \quad (1)$$

Формулы интересны тем, что они не привязаны к критериям качества диагностирования, а оперируют лишь двумя компонентами, таблицей ФН и вектором экспериментальной проверки. Выполнение процедуры диагностирования по формулам (1) для вектора экспериментальной проверки $m_1 = (0101010010010)$, заданного в последней таблице ФН, дает

результат: $B^s(m_1, T) = D_2$, который не хуже, чем ранее полученный методом анализа столбцов. Для вектора экспериментальной проверки $m_2 = (1110011100000)$ результат

диагностирования имеет вид: $B^m(m_2, T) = L_1 \vee L_2$. Вычислительная сложность метода

анализа строк определяется следующей зависимостью: $Z^c = n^2$; $Z^r = n$. Первая оценка предназначена для подсчета числа координатных операций, вторая определяет вычислительную сложность процесса обработки на основе регистровых параллельных операций. Предложенные методы диагностирования уязвимостей для КС есть один из наиболее существенных компонентов инфраструктуры сервисного обслуживания проектируемых и функционирующих КС.

4. 3D-метод диагностирования уязвимостей

Мотивация определяется рыночной привлекательностью матричного метода поиска ФН в компонентах КС как самого технологичного, который ориентирован на параллельную обработку данных, что дает возможность существенно уменьшить время диагностического обслуживания при возникновении уязвимостей (проникновений).

Цель исследования – создание модели, метода и их программно-аппаратной реализации, ориентированных на существенное уменьшения времени тестирования и затрат памяти для хранения матрицы диагностирования путем формирования тернарных отношений (тест-монитор-функциональный компонент).

Задачи: 1) Разработка модели КС в виде транзакционного графа, а также матрицы активизации функциональных компонентов на тестах относительно выбранного множества

мониторов. 2) Разработка метода анализа матрицы активизации для поиска уязвимостей с заданной глубиной. 3) Синтез логических функций для встроенного диагностирования ФН.

Модель тестирования киберсистемы представлена в виде следующего преобразования начального уравнения диагноза, определенного хог-отношением параметров <тест – функциональность – уязвимые блоки>:

$$T \oplus F \oplus B = 0 \rightarrow B = T \oplus F \rightarrow B = \{T \times A\} \oplus F \rightarrow B = \{T \times A\} \oplus \{F \times m\},$$

которое оформлено в тернарное матричное отношение компонентов:

$$M = \{\{T \times A\} \times \{B\}\} \leftarrow M_{ij} = (T \times A)_i \oplus B_j.$$

Здесь координата матрицы (таблицы) равна 1, если пара тест-монитор $(T \times A)_i$ проверяет (активизирует) ФН функционального блока $B_j \in B$.

Модель киберсистемы представлена в виде транзакционного графа $G = \langle V, A \rangle$, $V = \{B_1, B_2, \dots, B_i, \dots, B_n\}$, $A = \{A_1, A_2, \dots, A_j, \dots, A_m\}$, где определены множество дуг – функциональных блоков и вершин – мониторов для наблюдения совокупности примитивов КС. Для целей диагностирования на графовую модель накладывается совокупность тестовых сегментов $T = \{T_1, T_2, \dots, T_r, \dots, T_k\}$, которая активизирует транзакционные пути в графе. В общем случае модель тестирования представлена декартовым произведением $M = \langle V \times A \times T \rangle$, которая имеет размерность $Q = n \times m \times k$. Чтобы уменьшить объем диагностической информации каждому тесту предлагается поставить в соответствие монитор, который отвечает за визуализацию пути активизации функциональных блоков, что дает возможность уменьшить размерность модели (матрицы) до $Q = n \times k$ при сохранении всех возможностей отношения триады $M = \langle V \times A \times T \rangle$. Для пары тест-монитор возможны взаимно-однозначные соответствия $\langle T_i \rightarrow A_j \rangle$, функциональные $\langle \{T_i, T_r\} \rightarrow A_j \rangle$ и инъективные $\langle T_i \rightarrow \{A_j, A_s\} \rangle$. Такое многообразие соответствий дает возможность дублировать один тестовый сегмент для различных мониторов, равно как и нагружать несколько тестов на один и тот же монитор. При этом ячейка матрицы $M_{ij} = \{0, 1\}$ всегда сохраняет свою размерность, равную одному биту.

Аналитическая обобщенная модель матричного диагностирования с использованием механизма мониторов ориентирована на достижение заданной глубины поиска дефектов и представлена в следующем виде:

$$M = f(G, L, T, B, A, t); \quad B = \{B_1, B_2, \dots, B_i, \dots, B_m\}; \quad L = \{L_1, L_2, \dots, L_i, \dots, L_n\};$$

$$A(t) = \{A_1, A_2, \dots, A_i, \dots, A_k\}; \quad A \subseteq L; \quad G = L \times B; \quad k \leq n; \quad T = \{T_1, T_2, \dots, T_i, \dots, T_p\}.$$

Здесь B_i – группа операторов кода, нагруженная на вершину L_i (блок КС) и формирующая ее состояние; G – функциональность, представленная транзакционным графом $G = (L, A) \times B$ в виде декартова произведения множества вершин и дуг; A – совокупность мониторов, как подмножество вершин транзакционного графа $A \subseteq L$. Метод поиска уязвимостей функциональных блоков (ФБ) использует предварительно построенную таблицу (матрицу) активизации ТАФБ $M = [M_{ij}]$, где строка есть отношение между тестовым сегментом и подмножеством активизированных блоков $T_i \rightarrow A_j \approx (M_{i1}, M_{i2}, \dots, M_{ij}, \dots, M_{in}), M_{ij} = \{0, 1\}$, наблюдаемых на мониторе A_j . Столбец таблицы формирует отношение между функциональным блоком, тестовыми сегментами и мониторами $M_j = B_j \approx f(T, A)$. В механизм мониторов может быть введен параметр модельного времени, который частично усложняет матрицу активизации, указывая временной или модельный такт, на котором выполняется мониторинг состояния вершины или функционального блока на тест-сегменте $A_j = f(T_i, B_j, t_j)$.

Для диагностирования неисправностей на стадии моделирования определяется обобщенная реакция (вектор-столбец) $m = \{m_1, m_2, \dots, m_i, \dots, m_p\}$ механизма мониторов A на тест-сегменты T путем формирования $m_i = f(T_i, A_i)$. Поиск уязвимости ФБ основан на определении хог-операции между вектором состояния ассерций и столбцов таблицы ФН $m \oplus (M_1 \vee M_2 \vee \dots \vee M_j \vee \dots \vee M_n)$. Решение выбирается методом хог-анализа столбцов, путем выбора совокупности векторов B_j с минимальным числом единичных координат

$B = \min_{j=1, n} [B_j = \sum_{i=1}^p (B_{ij} \oplus m_i)]$, формирующих функциональные блоки с уязвимостями, проверяемыми на тестовых сегментах. В дополнение к модели матричного диагностирования необходимо описать следующие важные свойства матрицы:

- 1) $M_i = (T_i - A_j)$;
- 2) $\bigvee_{i=1}^m M_{ij} \rightarrow \forall M_j = 1$;
- 3) $M_{ij} \oplus M_{rj} \neq M_{ij}$;
- 4) $M_{ij} \oplus M_{ir} \neq M_{ij}$;
- 5) $\log_2 n \leq k \leftrightarrow \log_2 |B| \leq |T|$;
- 6) $B_j = f(T, A) \rightarrow B \oplus T \oplus A = 0$.

Свойства означают: 1) Каждая строка матрицы есть соответствие или подмножество декартова произведения (тест-монитор). 2) Дизъюнкция всех строк матрицы дает вектор, равный единицам по всем координатам. 3) Все строки матрицы различны, что исключает тестовую избыточность. 4) Все столбцы матрицы различны, что исключает существование эквивалентных неисправностей. 5) Число строк матрицы должно быть больше двоичного логарифма от числа столбцов, что определяет потенциальную диагностируемость всех ФН блоков. 6) Функция диагностирования блока с ФН зависит от совокупного теста и мониторов, которые должны быть минимизированы без нарушения диагнозопригодности.

Для пояснения работоспособности модели и метода рассмотрим функциональности трех модулей, входящих в примера КС. Первым является компонент Row_buffer, для которого создан транзакционный граф (рис. 4). Вершины представлены состояниями переменных и мониторов, отвечающих за входящие в вершину транзакции или дуги, которым соответствуют функциональные блоки.

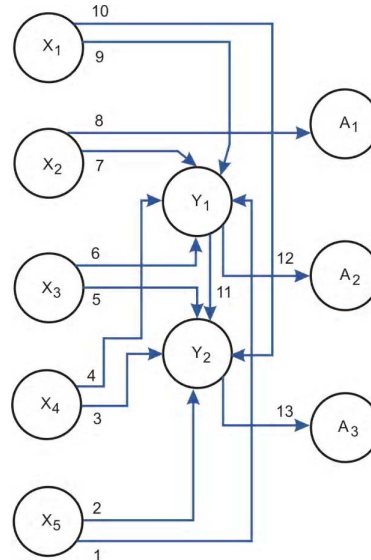


Рис. 4. Компонент Row_buffer транзакционного графа

На основе графа, полученного в процессе анализа КС, строится таблица активизации функциональных блоков, строки которой представляют пути активизации блоков к заказанной вершине-монитору. Таблица представляет собой покрытие строками-путями всех столбцов или функциональных блоков. При этом в ней не должно быть хотя бы двух одинаковых столбцов. Отличие таблицы заключается в формировании пары <тест-наблю-

даемая вершина>, что дает возможность существенно сократить ее размерность при 100% распознавании всех дефектных блоков. Здесь самое главное отличие предложенной модели заключается в возможности описания с помощью таблицы следующих отношений: различные тесты – одна вершина; один тест – различные вершины:

A _{ij}	T ₁	T ₂	T ₃	T ₄	T ₅	T ₆	T ₇	T ₈	T ₉	T ₁₀	T ₁₁	T ₁₂	T ₁₃
t ₁ → D ₃	.	.	.	.	.	.	.	1	.	.	.	.	.
t ₂ → D ₁	1	.	.	.	.	.	.	.	.	.	.	.	1
t ₃ → D ₁	.	.	1	.	.	.	.	.	.	.	.	.	1
t ₄ → D ₁	.	.	.	.	1	.	.	.	.	.	.	.	1
t ₅ → D ₁	.	.	.	.	.	.	.	.	.	.	1	.	1
t ₆ → D ₁	.	.	.	.	.	.	.	.	1	.	.	.	1
t ₇ → D ₂	.	1	.	.	.	.	.	.	.	.	.	1	.
t ₈ → D ₂	.	.	.	1	.	.	.	.	.	.	.	1	.
t ₉ → D ₂	.	.	.	.	.	1	.	.	.	.	.	1	.
t ₁₀ → D ₂	.	.	.	.	.	.	1	.	.	.	.	1	.
t ₁₁ → D ₂	.	.	.	.	.	.	.	.	.	1	.	1	.

С помощью матрицы активизации функциональных блоков (транзакционного графа) и хог-метода поиска дефектов достаточно просто синтезировать логические функции для формирования комбинационной схемы, определяющей в процессе и по результатам моделирования номер функционального блока, который имеет семантические ошибки:

$$D_3 = T_8^1;$$

$$D_1 = T_{13}^1 T_1^1 \vee T_{13}^1 T_3^1 \vee T_{13}^1 T_5^1 \vee T_{13}^1 T_{11}^1 \vee T_{13}^1 T_9^1;$$

$$D_2 = T_{12}^1 T_2^1 \vee T_{12}^1 T_4^1 \vee T_{12}^1 T_6^1 \vee T_{12}^1 T_7^1 \vee T_{12}^1 T_{10}^1.$$

Такое свойство становится возможным благодаря отсутствию эквивалентных уязвимостей или одинаковых столбцов в матрице активизации. Поэтому фиксация фактического состояния всех мониторов в вершинах D₁, D₂, D₃ на 11 тестовых наборах дает возможность однозначно идентифицировать некорректный функциональный модуль путем выполнения хог-операции между вектором ассерций и столбцами матрицы активизации. Нулевое значение всех координат результата хог-операции определяет номер столбца, соответствующего уязвимости модуля. Имплементация модели и метода в логическую функцию позволяет возможность определять уязвимый блок еще до завершения диагностического эксперимента, если это возможно. Это означает существенную экономию времени диагностирования отдельных видов уязвимостей. Например, тест-монитор t₁ → D₃ дает возможность идентифицировать уже на первом тесте уязвимость блока B₈.

В качестве второго тестового примера для практического использования разработанной модели активизации и хог-метода поиска дефектов далее предлагается синтез матрицы диагностирования для модуля, представленного на рис. 5.

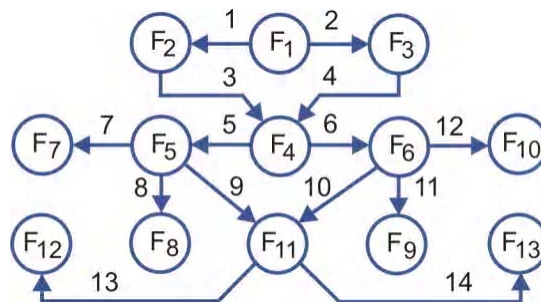


Рис. 5. Транзакционный граф main-TL

Данному графу ставится в соответствие следующая матрица диагностирования:

A_{ij}	T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8	T_9	T_{10}	T_{11}	T_{12}	T_{13}	T_{14}
$P_1 \rightarrow F_7$	1	.	1	.	1	.	1	.	.	.	.	.	.	.
$P_2 \rightarrow F_8$	.	1	.	1	1	.	.	1	.	.	.	.	.	.
$P_3 \rightarrow F_9$	1	.	1	.	.	1	.	.	.	.	1	.	.	.
$P_4 \rightarrow F_{10}$	.	1	.	1	.	1	.	.	.	.	.	1	.	.
$P_5 \rightarrow F_{12}$	1	.	1	.	1	.	.	.	1	.	.	.	1	.
$P_6 \rightarrow F_{13}$	.	1	.	1	.	1	.	.	.	1	.	.	.	1
$P_1 \rightarrow F_2$	1	.	.	.	.	.	.	.	.	.	.	.	.	.
$P_2 \rightarrow F_3$	.	1	.	.	.	.	.	.	.	.	.	.	.	.

Система функций диагностирования имеет такой вид:

$$F_7 = T_1^1 T_3^1 T_5^1 T_7^1; F_8 = T_2^1 T_4^1 T_5^1 T_8^1; F_9 = T_{11}^1 T_6^1 T_1^1 T_3^1;$$

$$F_{10} = T_4^1 T_5^1 T_6^1 T_{12}^1; F_{12} = T_1^1 T_3^1 T_5^1 T_9^1 T_{13}^1; F_{13} = T_2^1 T_4^1 T_6^1 T_{10}^1 T_{14}^1;$$

$$F_2 = T_1^1; F_3 = T_2^1.$$

5. Мультиуровневая модель и метод диагностирования ФН

Представлена мультидеревом B , где каждая вершина есть трехмерная таблица активизации функциональных модулей, а дуги, исходящие из нее, есть переходы на нижний уровень детализации при диагностировании, когда замена рассматриваемого уязвимого функционального блока слишком дорога:

$$B = [B_{ij}^{rs}], \text{ card } B = \sum_{r=1}^n \sum_{s=1}^{m_r} \sum_{j=1}^{k_{rs}} B_{ij}^{rs},$$

n – число уровней мультидерева диагностирования; m_r – количество функциональных

блоков или компонентов на уровне r ; k_{rs} – число компонентов в таблице B_{ij}^{rs} ; $B_{ij}^{rs} = \{0,1\}$

– компонент таблицы активизации, определенный сигналами проверки (непроверки) функ-

ционального модуля тест-сегментом T_{i-A_i} относительно наблюдаемого монитора A_i .

Каждая вершина-таблица имеет число исходящих вниз дуг, равное количеству функциональных блоков, диагностируемых (представленных) в таблице активизации. Структура мультидерева, соответствующая модели многоуровневого диагностирования, представлена на рис. 6.

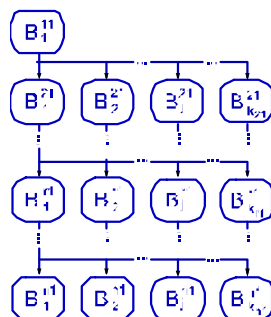


Рис. 6. Фрагмент мультидерева диагностирования

Процесс-модель или метод поиска дефектов по мультидереву диагностирования сводится к созданию движка (рис. 7) для спуска по одной из ветвей дерева на такую глубину, которая удовлетворяет пользователя по степени детализации:

$$B_j^{rs} \oplus A^{rs} = \begin{cases} 0 \rightarrow \{B_j^{r+1,s}, R\}; \\ 1 \rightarrow \{B_{j+1}^{rs}, T\}. \end{cases}$$

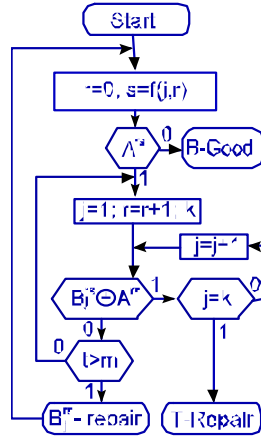


Рис. 7. Движок обхода мультидерева диагностирования

Здесь выполняется векторная хог-операция между столбцами матрицы и вектором экспериментальной проверки A^{rs} , который определяется реакцией функциональности, снятой с мониторов (ассерций) при подаче всех тест-сегментов. Если хотя бы одна координата полученной векторной хог-суммы равна нулю $B_j^{rs} \oplus A^{rs} = 0$, то выполняется одно из действий: переход к матрице активизации нижнего уровня $B_j^{r+1,s}$ или восстановление работоспособности (устранение уязвимости) функционального блока B_j^{rs} . При этом анализируется, что важнее: 1) время – тогда устранение уязвимости рассматриваемого блока; 2) деньги – тогда осуществляется переход вниз, для уточнения места уязвимости, поскольку замена более мелкого блока существенно уменьшает стоимость восстановления неустойчивости. Если хотя бы одна координата полученного вектора хог-суммы равна единице $B_j^{rs} \oplus A^{rs} = 1$, то выполняется переход к анализу следующего столбца матрицы. При нулевых значениях всех координат вектора (ассерционных) мониторов $A^{rs} = 0$ фиксируется неустойчивое состояние всего изделия. Если в рассматриваемой таблице зафиксированы все векторные хог-суммы, не равные нулю $B_j^{rs} \oplus A^{rs} = 1$, то коррекции подлежит тест, построенный для проверки данной функциональности.

Технологическая модель инфраструктуры встроенного тестирования, диагностирования и восстановления неустойчивости (рис. 8) имеет три компонента: 1. Тестирование КС (Unit Under Test (UUT)) с использованием эталонной модели (Model Under Test (MUT)) для формирования вектора экспериментальной проверки m_a , размерность которого соответствует числу тестовых наборов. 2. Поиск дефектов на основе анализа таблицы уязвимостей А. 3. Восстановление неустойчивости КС посредством замены уязвимых блоков на компоненты из Spare Good Primitives.

Процесс-модель встроенного сервисного обслуживания работает в реальном масштабе времени и позволяет поддерживать в работоспособном состоянии, без вмешательства человека, КС, что является целесообразным решением в случае применения технологий,

Таким образом, представленная на рисунке граф-схема дает возможность эффективно осуществлять сервисное обслуживание сколь угодно сложной киберсистемы. Преимущества такого движка, инвариантного к уровням иерархии, заключаются в простоте подготовки и представления диагностической информации в виде минимизированной таблицы активизации функциональных блоков или деструктивных компонентов (уязвимостей) на тестовых сегментах.

В последнем случае эффект (уменьшение времени) получен благодаря введению дополнительной инфраструктуры (рис. 9) к функциональности проекта, позволяющей избирательно осуществлять тестирование и диагностирование, а также перепрограммировать отдельные модули в случае фиксации уязвимостей.

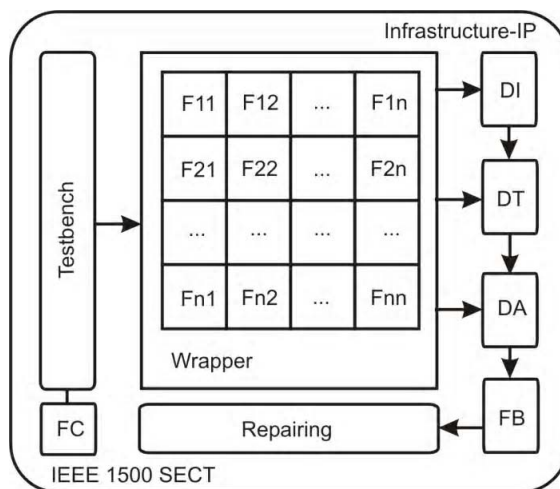


Рис. 9. Инфраструктура тестирования КС

Здесь представлены блоки: Testbench – тесты проверки функциональных блоков, FC – функциональное покрытие теста, F – функциональные блоки КС, DI – диагностическая информация, DT – методы и средства диагностирования, DA – результаты анализа процесса диагностирования, FB – уязвимые функциональные модули, Repairing – восстановление работоспособности функциональных модулей. Сервисное обслуживание отдельной ячейки функциональности осуществляется с помощью ячейки граничного сканирования, представленной на рис. 10.

6. Выводы

Предложена усовершенствованная процесс-модель определения уязвимостей в КС, которая отличается использованием хог-операции, что дает возможность повысить быстродействие диагностирования одиночных или кратных ФН на основе параллельного анализа таблицы ФН с помощью логических векторных операций and, or, хог.

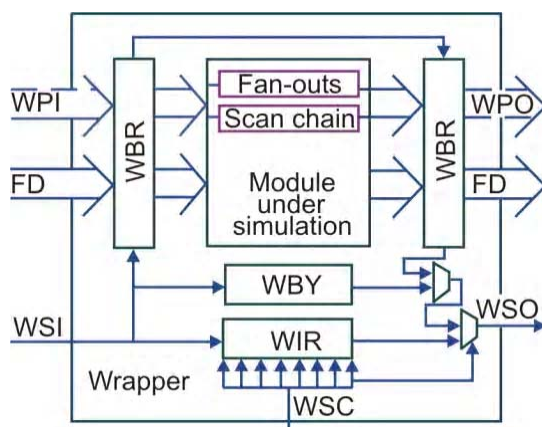


Рис. 10. Ячейка граничного сканирования

Разработаны структурная модель в виде мультидерева и метод (движок) его быстрого обхода, которые отличаются инвариантностью к уровням иерархии КС для диагностирования уязвимостей на тестовых сегментах, что дает возможность эффективно, в реальном масштабе времени, осуществлять сервисное обслуживание сколь угодно сложной киберсистемы.

Список литературы: 1. Бондаренко М.Ф., Хаханов В.И., Литвинова Е.И. Структура логического ассоциативного мультипроцессора // Автоматика и телемеханика. 2012. № 10. С. 71-92. 2. Hahanov V., Wajeb Gharibi, Litvinova E., Chumachenko S. Information analysis infrastructure for diagnosis // Information an international interdisciplinary journal. 2011. Japan. Vol.14, № 7. P. 2419-2433. 3. Bishop M. About Penetration Testing // IEEE Security & Privacy. 2007. Vol. 5, Iss. 6. P. 84 – 87. 4. Mainka C., Somorovsky J., Schwenk J. Penetration Testing Tool for Web Services Security // 2012 IEEE Eighth World Congress on Services (SERVICES). 2012. P. 163 – 170. 5. Salas P.A.P., Padmanabhan Krishnan, Ross K.J. Model-based Security Vulnerability Testing // 18th Australian Software Engineering Conference. 2007. P. 284 – 296. 6. Bau Jason, Bursztein Elie, Gupta Divij, Mitchell John. State of the Art: Automated Black-Box Web Application Vulnerability Testing // 2010 IEEE Symposium on Security and Privacy. 2010. P. 332 – 345. 7. Shahriar H., Zulkernine M. Automatic Testing of Program Security Vulnerabilities // 33rd Annual IEEE International Computer Software and Applications Conference. 2009. Vol. 2. P. 550 – 555. 8. Sedaghat S., Adibniya F., Sarram M.-A. The investigation of vulnerability test in application software // International Conference on the Current Trends in Information Technology (CTIT). 2009. P.1 – 5. 9. Wilhelm T. Professional Penetration Testing. Syngress. 2009. 524 p. 10. Shakeel A., Heriyanto T. BackTrack 4: Assuring Security by Penetration Testing. Packt Publishing. 2011. 392 p. 11. Хаханов В.И., Anders Carlsson, Чумаченко С.В. Инфраструктура PenTestING и управления уязвимостью // АСУ и приборы автоматики. 2012. Вып. 160. С. 36-54.

Поступила в редколлегию 17.10.2012

Хаханов Владимир Иванович, д-р техн. наук, декан факультета КИУ, профессор кафедры АПВТ ХНУРЭ. Научные интересы: техническая диагностика цифровых систем, сетей и программных продуктов. Увлечения: баскетбол, футбол, горные лыжи. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-326. E-mail: hahanov@kture.kharkov.ua.

Anders Carlsson, COM School of Computing, Blekinge Institute of Technology (BTH). Address: Room H454D, Blekinge Institute of Technology, SE-37179 Karlskrona, Sweden.

Чумаченко Светлана Викторовна, д-р техн. наук, профессор кафедры АПВТ ХНУРЭ. Научные интересы: математическое моделирование, теория рядов, методы дискретной оптимизации. Увлечения: путешествия, любительское фото. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-326. E-mail: ri@kture.kharkov.ua.

Бутенко Сергей Александрович, студент группы КИ-09-4 факультета КИУ. Научные интересы: диагностика цифровых систем и сетей. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-326.

АНАЛИЗ СИСТЕМ АВТОМАТИЗИРОВАННОГО МОНИТОРИНГА АВТОМОБИЛЬНОГО ТРАНСПОРТА И УПРАВЛЕНИЯ ДОРОЖНЫМ ДВИЖЕНИЕМ

Анализируются наиболее масштабные проекты, направленные на реализацию автоматизированного управления дорожным движением и мониторинга автомобильного транспорта. Рассматриваются как уже реализованные в реальных масштабах проекты управления и мониторинга OnStar, NEXCO Central и ECall Japan (США, Канада, Япония), так и разрабатываемые в настоящий момент системы «ЭРА-ГЛОНАСС» и ECall Europe (Евросоюз, Российская Федерация).

Введение

На сегодняшний день актуальность систем автоматизированного управления дорожным движением непрерывно возрастает. Количество автомобилей на дорогах увеличивается, а с ним – количество аварий, травм и смертей. Правительства многих стран озабочены проблемами регулирования, контроля и мониторинга дорожного движения. В связи с глобальным переходом корпораций и пользователей в облака чрезвычайно актуальной и рыночно привлекательной становится проблема защиты информации и компонентов киберпространства от несанкционированного доступа, деструктивных проникновений, вирусов. Возникает необходимость создания надежной, тестопригодной и защищенной от несанкционированных проникновений инфраструктуры киберпространства и его компонентов (виртуальные персональные компьютеры и корпоративные сети) по аналогии с существующими сегодня решениями в реальном кибернетическом мире. Таким образом, каждый сервис, разрабатываемый в реальном мире, должен быть помещен в соответствующую ячейку облака, которое объединяет близкие по функциональности и полезные человеку компоненты. В США, Канаде и Японии уже реализованы масштабные проекты, призванные решить создавшиеся проблемы. Это специализированные системы автоматизированного мониторинга дорожного движения OnStar [7], NEXCO Central [8] и ECall Japan [2]. В настоящий момент Российская Федерация и Евросоюз занимаются разработкой спутниковых систем навигации и контроля дорожного движения – «ЭРА-ГЛОНАСС» [6] и ECall Europe [1, 3-5].

Данная работа раскрывает основные преимущества и недостатки каждой из систем, что позволяет принять правильные интеграционные архитектурные решения для создания абсолютно новой системы мониторинга дорожного движения и управления транспортом. Задача исследования состоит в определении необходимости, возможности и неизбежности создания интеллектуального облачного сервиса мониторинга и управления дорожным движением, оптимизирующего исполнение транспортных маршрутов всеми участниками дорожного движения.

Целью данного исследования является выявление основных достоинств и недостатков в архитектуре, эргономике и принципиальной схеме функционирования каждой из систем, анализ существующих и разрабатываемых решений.

1. Система мониторинга транспортных средств OnStar

Система разработана компанией OnStar Corporation, основанной в 1995 году. В самом начале становления компании продавались отдельные устройства, которые были доступны только владельцам определенных моделей автомобилей: Cadillac DeVille, Cadillac Seville и Cadillac Eldorado. К 2005 году OnStar сервисы стали доступны владельцам автомобилей нескольких наиболее распространенных марок: Acura, Audi, Isuzu, Subaru, Volkswagen. Стоит отметить, что к этому времени устройства поступили в серийное производство и устанавливались в автомобили на этапе сборки. В апреле 2006 года уже насчитывалось 500 000 пользователей системы. В 2009 году компания вышла на Китайский рынок. На данный момент насчитывается около 4 000 000 пользователей данного сервиса [7]. Стоимость одного устройства мониторинга порядка \$ 200.

Рассмотрим подробнее технологии, которые используются для мониторинга транспортных средств. Система использует CDMA канал связи, предоставляемый преимущественно Verizon Wireless в США и Bell Mobility в Канаде. Для определения местоположения используется GPS. Имеется возможность голосовой связи с операторами.

Информация с сенсоров (в основном это датчики ударов и срабатывания подушек безопасности) автоматически передается в call-центры. Это позволяет немедленно оповестить о местоположении аварии спасательные и правоохранительные органы. Кроме этого, все машины, оборудованные данной системой, имеют GPS передатчик, который позволяет отследить угнанный автомобиль. Также имеется возможность получения информации о скорости, расходе топлива и направлении движения. Это позволяет сделать выводы о стиле вождения автомобиля. Данная информация используется страховыми компаниями для расчета стоимости индивидуальных страховых полисов. Новые модели автомобилей оборудуются системой удаленной остановки двигателя. После такой остановки автомобиль можно завести только после ввода специального секретного кода.

Пользователи могут применять различные тарифные планы [7]:

1. *Safe & Sound*: 18.95\$ в месяц.

Тариф включает автоматическое оповещение об аварии, мониторинг угнанного автомобиля, аварийные сервисы (вызов эвакуатора, передвижной СТО и т. п.) в не зоны покрытия мобильной сети, а также удаленную диагностику транспортного средства (позволяет решить незначительные проблемы при помощи инструкций call-центра).

2. *Directions & Connections*: \$ 28.90 в месяц.

Тариф включает все перечисленные выше сервисы. Дополнительно имеется возможность мониторинга направления движения и стиля вождения автомобилиста.

Позволяет получить скидку при оформлении страхового полиса.

2. Система управления дорожным движением NEXCO Central

Система разработана Japan Highway Public Corporation. Принцип работы заключается в глобальном мониторинге дорожного движения на главных автострадах страны. На данный момент система покрывает порядка 2000 км дорог[8].

Система централизована. Центр управления движением находится в Токио (рис.1). Датацентр обрабатывает огромное количество данных, получаемых с дорожных датчиков с минутным интервалом. Это обеспечивает максимально реальную картину дорожной ситуации.



Рис. 1. Центр управления дорожным движением в Японии

На дорогах установлено 744 точки доступа, которые позволяют работать аварийным телефонным каналам и датчикам на передачу необходимой информации о дорожной ситуации [8].

Для передачи данных используется глобальная IP сеть, при помощи которой информация с датчиков поступает на мониторы центра управления движением.

Глобальная передача данных обеспечивается посредством оптоволоконных коммуникаций. Это позволяет быстро обрабатывать телефонные звонки и поступающие данные.

Эта система запущена параллельно с действующей в апреле 2012 года. Серверное пространство на 90% меньше, чем у старой системы, следовательно, наблюдается значительное энергосбережение [8].

Разработчики системы предлагают полный спектр услуг по разработке и внедрению подобной системы в других странах (рис.2).



Рис. 2. Предлагаемые услуги по разработке системы мониторинга

3. ECall Япония

С середины 1980-х гг. на всех дорогах страны была запущена интеллектуальная транспортная система, призванная осуществлять полную автоматизацию управления дорожным движением. На все автомобили стали устанавливать специальное бортовое навигационно-коммуникационное оборудование, с помощью которого обеспечивается контроль местоположения и состояния транспортного средства [1].

Передача информации и управляющих сигналов, а также дуплексная связь с водителем осуществляется диспетчерской службой быстрого реагирования под названием Ecall [4] (рис. 3).

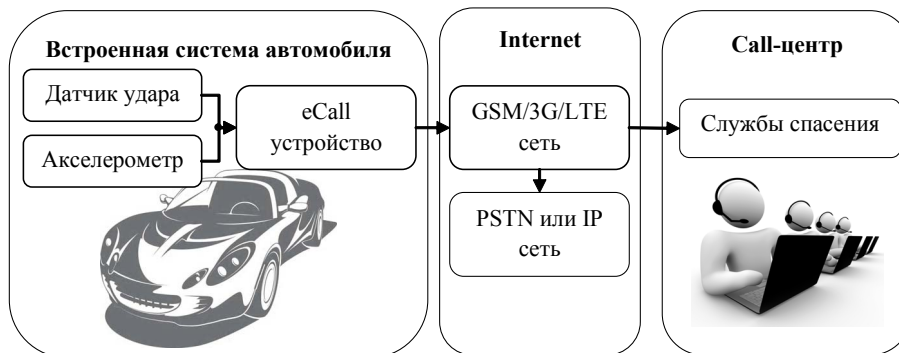


Рис. 3. Схема функционирования системы ECall

В результате успешной деятельности системы смертность на дорогах Японии значительно снизилась, в 2009 году составила 5 тыс. чел. [2] Власти страны планируют сократить число погибших до нуля.

4. Система глобального мониторинга транспорта «ЭРА ГЛОНАСС»

Система «ЭРА ГЛОНАСС» проектируется в соответствии с распоряжением Правительства РФ и предназначена для снижения уровня смертности и травматизма на дорогах за счет ускорения оповещения служб экстренного реагирования при авариях и других чрезвычайных ситуациях.

Система будет включать навигационно-телекоммуникационные терминалы, которые начнут массово устанавливаться на транспортные средства, начиная с 2013 года, и соответствующую инфраструктуру, охватывающую все субъекты РФ (рис.4). Для обеспечения работоспособности системы на территории РФ Правительство России зарезервировало специальные телефонные коды, которые будут использовать для связи с экстренными центрами приема информации [6].

Предполагается законодательно обязать всех автовладельцев установить бортовое оборудование к 2020 году, стоимость которого составит около 3 тыс. российских рублей. Плата за пользование системой взиматься не будет. «ЭРА ГЛОНАСС» будет совместима с аналогичными службами, которые будут созданы в странах Таможенного союза, а также Евросоюза, где с 2015 года должна заработать система eCall [6].

Система идентификации и мониторинга объектов дорожного движения основывается на технологии радиочастотной идентификации малого радиуса действия (RFID) в сочетании с системами спутниковой навигации ГЛОНАСС и GPS. Проект решает задачи мониторинга транспортных средств, удаленного контроля документов и идентификации объектов транспортной инфраструктуры.

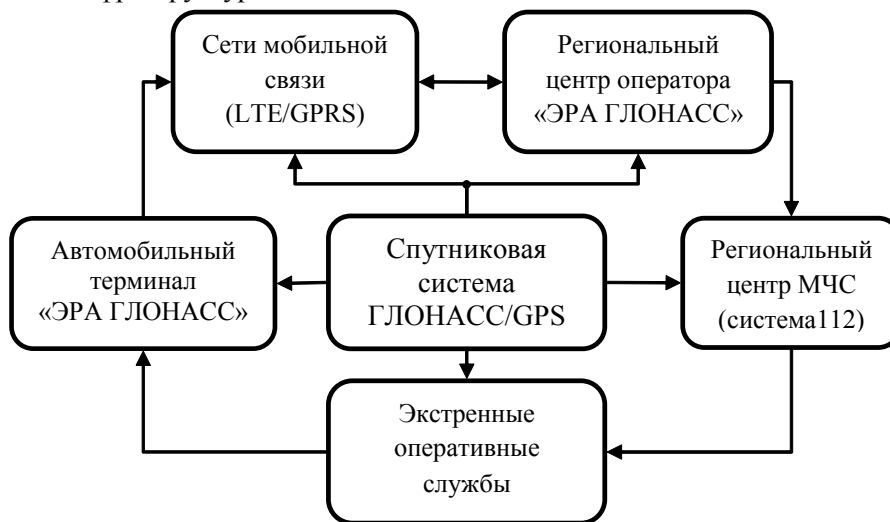


Рис. 4. Схема работы системы «ЭРА ГЛОНАСС»

Принцип работы системы заключается в следующем (см. рис.4) [6]. При тяжелой аварии, например, сопровождающейся срабатыванием подушек безопасности, терминал автоматически определяет координаты пострадавшего транспортного средства через спутники системы ГЛОНАСС, устанавливает связь с серверным центром системы мониторинга и передает данные об аварии по каналам сотовой связи оператору (рис.5). Оператор голосом уточняет детали происшествия и в случае подтверждения информации или при отсутствии ответа направляет службы экстренного реагирования. Водитель или пассажиры могут также вручную включить устройство, передать данные и связаться с оператором.

Система «ЭРА ГЛОНАСС», терминалы которой планируется устанавливать на все новые автомобили, позволяет оперативно передавать экстренным службам информацию о ДТП. В момент аварии терминал определяет географические координаты, используя навигационный сигнал системы ГЛОНАСС (а при необходимости и GPS), и передает их

вместе с дополнительной информацией в службу «112» через доступные сети операторов GSM-связи.



Рис. 5. Архитектура системы «ЭРА ГЛОНАСС»

Прогнозы аналитиков утверждают, что система может снизить количество ДТП в среднем на 5-10% и сократить наносимый экономике ущерб не менее, чем на 2% [6].

5. Система глобального мониторинга транспорта ECall Европа

С 2001 года странами Евросоюза также стала разрабатываться программа eCall, согласно которой в 2015 году весь автотранспорт, продаваемый на территории содружества, должен быть укомплектован навигационно-коммуникационными средствами, срабатывающими при аварии, после чего на номер 112 по каналам GSM-связи передается информация о местонахождении автомобиля на ближайший диспетчерский пункт (рис.6) [3].

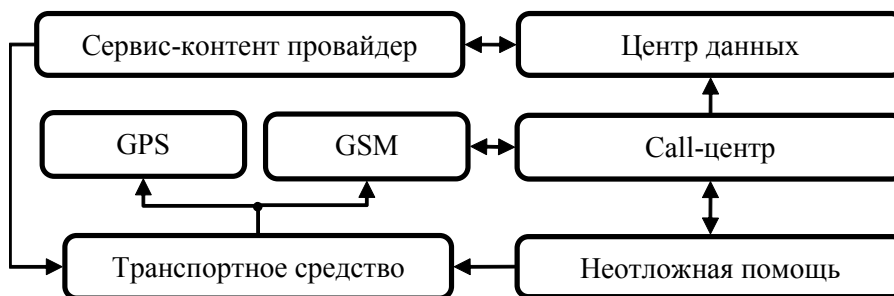


Рис. 6. Архитектура системы ECall

В Германии такими приборами уже с 2005 года стали оснащаться фуры грузоподъемностью свыше 12 т, в Швеции — грузовики массой свыше 3,5 тонны. Программа продвигается медленно из-за задержки развертывания спутниковой системы Galileo и бюрократических сложностей принятия совместных решений в рамках регламента ЕС.

Принцип работы системы заключается в следующем (см. рис.6) [6]. eCall автоматически активизируется, когда датчики, которые находятся внутри автомобиля, определяют аварию. Система набирает европейский экстренный номер 112, устанавливая телефонное соединение с ближайшим Call-центром, и отправляет детали происшествия в службы спасения, включая время происшествия, точное местоположение автомобиля и направление движения (особенно важного на дорогах и в туннелях). eCall может также быть включен вручную, с помощью нажатия кнопки в автомобиле, например, если кто-то стал свидетелем серьезной аварии [3].

По некоторым оценкам, eCall сможет ускорить время реагирования на экстренные ситуации на 40 % в городских регионах и на 50 % в сельской местности, и спасти до 2500 жизней в год. Меры, принятые 26 ноября 2012 года, устанавливают условия, согласно которым центры реагирования на экстренные вызовы будут способны правильно принимать и обрабатывать вызовы eCall 112[5].

6. Сравнительный анализ рассмотренных систем

Общий анализ перечисленных выше систем мониторинга дорожного движения позволяет понять, что на сегодняшний день не существует проекта, направленного на комплексное решение поставленной проблемы. Ни одна существующая и ни одна проектируемая система не предполагает наличия полного спектра услуг, направленного на повышение комфорта автомобилиста. Речь идет либо только о полном мониторинге автомобиля (OnStar, ECall, «ЭРА ГЛОНАСС»), либо о глобальном мониторинге трафика (NEXCO Central) на основных и самых оживленных магистралях страны. Особенностью систем «ЭРА ГЛОНАСС» и ECall является использование собственных спутниковых систем навигации – ГЛОНАСС и Galileo соответственно. Однако ничто не говорит о том, что применение системы GPS (OnStar) менее эффективно. Основным преимуществом спутниковых систем навигации является большое покрытие. Недостатки таких систем можно увидеть в тоннелях, подземных и многоуровневых магистралях. В таком случае необходимо введение наземных датчиков движения. Такое решение предлагает японская система NEXCO Central. Также преимуществом системы является отсутствие необходимости установки специальных устройств на автотранспорт, однако такой подход полностью исключает мониторинг отдельных транспортных средств. Основным недостатком данного подхода является ограниченное покрытие (только основные автомагистрали и трассы).

Услуги пользователям предоставляет только система США OnStar. Данная система ориентирована на обеспечение комфорта и безопасности как водителя и пассажиров, так и самого транспортного средства. Однако этим система и ограничена. OnStar не предоставляет какой-либо статистической информации о дорожной ситуации. Также нет понятия централизованного управления. Система ориентирована только на отдельные автомобили, а не на транспортный поток.

Выводы

Таким образом, можно сделать вывод, что наиболее эффективным решением будет интеграция рассмотренных систем в одной универсальной. Это позволит обеспечить автомобилиста полным спектром услуг, начиная от эффективной навигации заканчивая круглосуточным предоставлением информации об автотранспортном средстве. Необходимо разработка интегрированной системы, направленной как на мониторинг каждого автомобиля в отдельности, так и на контроль дорожного движения в целом. Именно это и предлагает система мониторинга Green Wave Traffic, которая дает возможность автоматизировать процессы оптимального управления транспортными средствами и дорожным движением в режиме реального времени для решения социальных, гуманитарных, экономических и экологических проблем. Предлагаемая интеллектуальная система (инфраструктура, транспорт, облако) мониторинга и управления дорожным движением отличается от существующих структурной интеграцией трех взаимосвязанных интерактивных компонентов: 1) Существующие сервисы электронной картографии со средствами радиолокации и радионавигации. 2) Новый облачный сервис мониторинга и управления дорожным движением на основе дорожных контроллеров. 3) Усовершенствованные средства радиочастотной идентификации автомобиля и доступа к облачным сервисам для комфортного и безопасного передвижения по маршруту, оптимизации временных и материальных затрат. Научная новизна проекта определяется системной интеграцией облака мониторинга и управления, блоков радиочастотной идентификации транспорта, а также средств мониторинга и управления дорожной инфраструктуры.

Список литературы: 1. *Filjar. R.* ECall: Automatic notification of a road traffic accident / K. Vidovic, P. Britvic, M. Rimac // MIPRO. 2011. С. 600-605. 2. *SY Hu.* Simplified Design of the Embedded e-Call System Using GPS / TS Wey, MH Lin, NT Hu A// Display technology. 2011. С. 45-53. 3. *Pinart Carolina.* ECall-compliant early crash notification service for portable and nomadic devices / J. Carlos Calvo, Laura Nicholson, Josй A. Villaverde// MIT. 2011. С. 134-146. 4. *Werner Marc.* Cellular In-Band Modem Solution for eCall

Emergency Data Transmission / Christian Pietsch, Christoph Joetten // Communication. 2009. С. 198-207. **5.** Kohn Andreas. The eCall Program: Overview and Design Considerations // Sierra Wireless. 2010. P. 157-167p. **6.** http://www.nis-glonass.ru/projects/era_glonass [Электронный ресурс]. **7.** <https://www.onstar.com/web/portal/termsconditions> [Электронный ресурс]. **8.** <http://media.brintex.com/Occurrence/27/Brochure/1544/brochure.pdf> [Электронный ресурс].

Поступила в редколлегию 24.11.2012

Мизь Владимир Александрович, магистрант кафедры АПВТ ХНУРЭ. Научные интересы: обработка видео-изображений. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. +380 (95) 302-24-92. E-mail: vladimir-miz@yandex.ru

Хаханова Анна Владимировна, канд. техн. наук, доц. кафедры АПВТ ХНУРЭ. Научные интересы: сжатие и восстановление двоичных данных. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-326. E-mail: anna_hahan@mail.ru

УДК 625.8 : 624.048

В.А. БОГОМОЛОВ, С.Н. ИЕВЛЕВА, И.Л. РАЗНИЦЫН, М.В. СИДОРОВ

ЧИСЛЕННЫЙ АНАЛИЗ НАПРЯЖЕННО-ДЕФОРМИРОВАННОГО СОСТОЯНИЯ СЛОЯ ДОРОЖНОЙ ОДЕЖДЫ КАК ЛИНЕЙНОЙ И ГЕОМЕТРИЧЕСКИ-НЕЛИНЕЙНОЙ ВЯЗКОУПРУГОЙ СРЕДЫ НА ОСНОВании РЕОЛОГИЧЕСКОЙ ОДНОЭЛЕМЕНТНОЙ МОДЕЛИ КЕЛЬВИНА

Рассматривается задача математического моделирования слоя дорожных одежд как линейной и геометрически-нелинейной вязкоупругих сред. На основании методов теории вязкоупругости с использованием одноэлементной модели Кельвина строится уравнение баланса – аналог принципа виртуальных перемещений теории упругости. Для численного анализа построенной математической модели предлагается использовать метод конечных элементов.

Введение

Актуальность исследования. Дорожная одежда является одной из наиболее распространенных и затратных строительных конструкций, содержащих слои из материалов с разными свойствами, на которые действуют нагрузки от транспортных средств и естественно-климатических факторов. В настоящее время при решении проблемы расчета напряженно-деформированного состояния слоев дорожных одежд используются только методы, основанные на осесимметричной постановке задач теории упругости. На основании этой теории разработаны действующие ныне дорожно-строительные нормы для Украины, России и Белоруссии [1 – 3, 6]. Во многих зарубежных методиках дорожная одежда также рассматривается как многослойная упругая среда. Кроме того, математическая постановка задачи предусматривает, что дорожная одежда интерпретируется как полупространство или плита в полупространстве [8, 9, 11]. Таким образом, существующие методики расчета характеристик дорожных одежд построены в дополнительных предположениях наличия осевой симметрии в материале и в нагрузках, а также не учитывают наличия вязкости и нелинейные свойства материала, что в свою очередь не позволяет в общем случае определить полные тензоры напряжений и деформаций. Кроме того, существующие методы не позволяют рассчитать и спрогнозировать образование колеи. С учетом сказанного актуальной является задача построения математической модели дорожной одежды как вязкоупругой среды и разработка новых численных методов для её анализа.

Цели и задачи исследования. Целью настоящего исследования является разработка математической модели и нового метода численного анализа дорожных одежд как линейной и геометрически нелинейной вязкоупругой среды на основании одноэлементной модели Кельвина. Для этого необходимо решить следующие задачи:

– построить на основании принципа возможных перемещений математическую модель слоя дорожных одежд как вязкоупругой среды (рассмотреть отдельно линейную и геометрически-нелинейную постановки);

Emergency Data Transmission / Christian Pietsch, Christoph Joetten // Communication. 2009. С. 198-207. **5.** Kohn Andreas. The eCall Program: Overview and Design Considerations // Sierra Wireless. 2010. P. 157-167p. **6.** http://www.nis-glonass.ru/projects/era_glonass [Электронный ресурс]. **7.** <https://www.onstar.com/web/portal/termsconditions> [Электронный ресурс]. **8.** <http://media.brintex.com/Occurrence/27/Brochure/1544/brochure.pdf> [Электронный ресурс].

Поступила в редколлегию 24.11.2012

Мизь Владимир Александрович, магистрант кафедры АПВТ ХНУРЭ. Научные интересы: обработка видео-изображений. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. +380 (95) 302-24-92. E-mail: vladimir-miz@yandex.ru

Хаханова Анна Владимировна, канд. техн. наук, доц. кафедры АПВТ ХНУРЭ. Научные интересы: сжатие и восстановление двоичных данных. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-326. E-mail: anna_hahan@mail.ru

УДК 625.8 : 624.048

В.А. БОГОМОЛОВ, С.Н. ИЕВЛЕВА, И.Л. РАЗНИЦЫН, М.В. СИДОРОВ

ЧИСЛЕННЫЙ АНАЛИЗ НАПРЯЖЕННО-ДЕФОРМИРОВАННОГО СОСТОЯНИЯ СЛОЯ ДОРОЖНОЙ ОДЕЖДЫ КАК ЛИНЕЙНОЙ И ГЕОМЕТРИЧЕСКИ-НЕЛИНЕЙНОЙ ВЯЗКОУПРУГОЙ СРЕДЫ НА ОСНОВании РЕОЛОГИЧЕСКОЙ ОДНОЭЛЕМЕНТНОЙ МОДЕЛИ КЕЛЬВИНА

Рассматривается задача математического моделирования слоя дорожных одежд как линейной и геометрически-нелинейной вязкоупругих сред. На основании методов теории вязкоупругости с использованием одноэлементной модели Кельвина строится уравнение баланса – аналог принципа виртуальных перемещений теории упругости. Для численного анализа построенной математической модели предлагается использовать метод конечных элементов.

Введение

Актуальность исследования. Дорожная одежда является одной из наиболее распространенных и затратных строительных конструкций, содержащих слои из материалов с разными свойствами, на которые действуют нагрузки от транспортных средств и естественно-климатических факторов. В настоящее время при решении проблемы расчета напряженно-деформированного состояния слоев дорожных одежд используются только методы, основанные на осесимметричной постановке задач теории упругости. На основании этой теории разработаны действующие ныне дорожно-строительные нормы для Украины, России и Белоруссии [1 – 3, 6]. Во многих зарубежных методиках дорожная одежда также рассматривается как многослойная упругая среда. Кроме того, математическая постановка задачи предусматривает, что дорожная одежда интерпретируется как полупространство или плита в полупространстве [8, 9, 11]. Таким образом, существующие методики расчета характеристик дорожных одежд построены в дополнительных предположениях наличия осевой симметрии в материале и в нагрузках, а также не учитывают наличия вязкости и нелинейные свойства материала, что в свою очередь не позволяет в общем случае определить полные тензоры напряжений и деформаций. Кроме того, существующие методы не позволяют рассчитать и спрогнозировать образование колеи. С учетом сказанного актуальной является задача построения математической модели дорожной одежды как вязкоупругой среды и разработка новых численных методов для её анализа.

Цели и задачи исследования. Целью настоящего исследования является разработка математической модели и нового метода численного анализа дорожных одежд как линейной и геометрически нелинейной вязкоупругой среды на основании одноэлементной модели Кельвина. Для этого необходимо решить следующие задачи:

– построить на основании принципа возможных перемещений математическую модель слоя дорожных одежд как вязкоупругой среды (рассмотреть отдельно линейную и геометрически-нелинейную постановки);

– описать алгоритм численного анализа построенной математической модели на основании метода конечных элементов.

Для решения поставленных задач использованы методы теории вязкоупругости и численные методы решения задач математической физики. Слой дорожных одежд представляется как тело в трехмерном пространстве, находящееся под действием объемных (массовых) сил, поверхностных сил и перемещений, заданных на поверхности тела.

В работе не рассматриваются вопросы существования и единственности поставленных краевых задач. Предполагается, что все задачи поставлены корректно в некоторых функциональных пространствах для заданных входных данных и все математические модели рассматриваются с точки зрения их алгоритмизации для дальнейшего решения на ЭВМ.

1. Постановка задачи и основные соотношения для одноэлементной модели Кельвина

Пусть тело Ω находится в равновесии под действием объемных (массовых) сил F_i , поверхностных сил t_i и перемещений, заданных на поверхности $S = \partial\Omega$ тела Ω . Пусть также заданы нагрузки и кинематические ограничения на перемещения тела. Под действием этих факторов в теле возникают перемещения u_i , деформации ε_{ij} и напряжения σ_{ij} , $i = 1, 2, 3$, $j = 1, 2, 3$.

Будем считать, что на части S_σ поверхности S заданы поверхностные нагрузки:

$$\sum_{j=1}^3 \sigma_{ij} n_j = t_i, \quad i = 1, 2, 3, \text{ на } S_\sigma;$$

а на части S_φ поверхности S заданы перемещения $\mathbf{u} = \bar{\mathbf{u}}$ на S_φ .

Здесь $S = S_\sigma \cup S_\varphi$, $\mathbf{n} = (n_1, n_2, n_3)^T$ – единичный вектор внешней к S_σ нормали, $\mathbf{u} = (u_1, u_2, u_3)^T$ – искомый, $\bar{\mathbf{u}} = (\bar{u}_1, \bar{u}_2, \bar{u}_3)^T$ – заданный векторы.

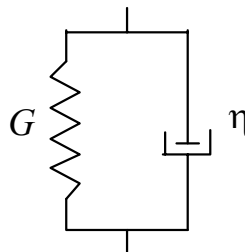
Уравнение равновесия деформированного тела получим на основании принципа возможных перемещений [7]

$$\iiint_{\Omega} \sum_{i=1}^3 \sum_{j=1}^3 \sigma_{ij} \delta \varepsilon_{ij} d\Omega = \iiint_{\Omega} \sum_{i=1}^3 F_i \delta u_i d\Omega + \iint_{S_\sigma} \sum_{i=1}^3 t_i \delta u_i ds, \quad (1)$$

где σ_{ij} – компоненты тензора напряжений Коши; $\delta \varepsilon_{ij}$ – вариации компонент тензора деформаций, отвечающие возможным перемещениям; F_i – компоненты вектора объемных сил, t_i – компоненты вектора поверхностных сил; δu_i – вариации компонент вектора возможных перемещений, $i, j = 1, 2, 3$.

Отметим, что на части S_φ поверхности S , где заданы перемещения, $\delta u_i = 0$, $i = 1, 2, 3$.

Для описания свойств материала дорожных одежд используем одноэлементную модель Кельвина (рисунок) [4].



Одноэлементная модель Кельвина

Для этой модели имеют место следующие соотношения:

$$\bar{\sigma}_{ij} = 2\eta \dot{\varepsilon}_{ij} + 2G \bar{\varepsilon}_{ij}, \quad (2)$$

$$\sigma_{cp.} = \frac{2\eta(1+\mu)}{1-2\mu} \left(\dot{\varepsilon}_{cp.} + \frac{G}{\eta} \varepsilon_{cp.} \right). \quad (3)$$

Здесь $\dot{\bar{\sigma}}_{ij} = \frac{d\bar{\sigma}_{ij}}{dt}$, $\dot{\bar{\varepsilon}}_{ij} = \frac{d\bar{\varepsilon}_{ij}}{dt}$, $\bar{\sigma}_{ij}$, $\bar{\varepsilon}_{ij}$ – компоненты девиаторной части тензоров напряжений и деформаций соответственно, $i, j=1, 2, 3$; $\sigma_{cp.}$, $\varepsilon_{cp.}$ – средние значения компонент шаровых тензоров напряжений и деформаций соответственно: $\sigma_{cp.} = \frac{1}{3}(\sigma_{11} + \sigma_{22} + \sigma_{33})$, $\varepsilon_{cp.} = \frac{1}{3}(\varepsilon_{11} + \varepsilon_{22} + \varepsilon_{33})$; G – модуль упругости; η – коэффициент вязкого сопротивления; μ – коэффициент Пуассона.

Компоненты полных тензоров напряжений и деформаций определяются соотношениями

$$\sigma_{ij} = \bar{\sigma}_{ij} + \delta_{ij}\sigma_{cp.}, \quad \varepsilon_{ij} = \bar{\varepsilon}_{ij} + \delta_{ij}\varepsilon_{cp.}, \quad i, j=1, 2, 3,$$

где $\delta_{ij} = \begin{cases} 1, & i=j, \\ 0, & i \neq j, \end{cases}$ – символ Кронекера.

С учетом формул (2) и (3) компоненты полного тензора напряжений запишутся в виде

$$\sigma_{ij} = \bar{\sigma}_{ij} + \delta_{ij}\sigma_{cp.} = 2\eta\dot{\bar{\varepsilon}}_{ij} + 2G\bar{\varepsilon}_{ij} + \delta_{ij}\frac{2\eta(1+\mu)}{1-2\mu} \left(\dot{\varepsilon}_{cp.} + \frac{G}{\eta} \varepsilon_{cp.} \right). \quad (4)$$

Для сокращения записи введем обозначение для дифференциального оператора

$$\mathcal{D} = 2\eta\frac{d}{dt} + 2G.$$

Тогда (4) приобретает вид

$$\sigma_{ij} = \mathcal{D}\bar{\varepsilon}_{ij} + \delta_{ij}\frac{1+\mu}{1-2\mu}\mathcal{D}\varepsilon_{cp.},$$

а с учетом того, что $\bar{\varepsilon}_{ij} = \varepsilon_{ij} - \delta_{ij}\varepsilon_{cp.}$, получим

$$\sigma_{ij} = \mathcal{D}\varepsilon_{ij} + \delta_{ij}\frac{3\mu}{1-2\mu}\mathcal{D}\varepsilon_{cp.}. \quad (5)$$

2. Построение математической модели слоя дорожной одежды как вязкоупругой среды (линейный случай) и применение метода конечных элементов для ее численного анализа

Для геометрически-линейных задач компоненты тензора деформаций определяются соотношениями Коши [7]

$$\varepsilon_{ij} = \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right), \quad i, j=1, 2, 3, \quad (6)$$

а вариация деформации определяется так:

$$\delta\varepsilon_{ij} = \frac{1}{2} \left(\frac{\partial \delta u_i}{\partial x_j} + \frac{\partial \delta u_j}{\partial x_i} \right), \quad i, j=1, 2, 3.$$

С учетом соотношений Коши (6) и (5) получим

$$\varepsilon_{cp.} = \frac{1}{3} \sum_{k=1}^3 \frac{\partial u_k}{\partial x_k},$$

$$\sigma_{ij} = \frac{1}{2} \left(\frac{\partial}{\partial x_j} \mathcal{D}u_i + \frac{\partial}{\partial x_i} \mathcal{D}u_j \right) + \delta_{ij} \frac{\mu}{1-2\mu} \sum_{k=1}^3 \frac{\partial}{\partial x_k} \mathcal{D}u_k, \quad i, j = 1, 2, 3. \quad (7)$$

Подставив (7) в (1), для неизвестных перемещений u_1, u_2, u_3 получим интегро-дифференциальное соотношение

$$\begin{aligned} \frac{1}{2} \iiint_{\Omega} \sum_{i=1}^3 \sum_{j=1}^3 \left[\frac{1}{2} \left(\frac{\partial}{\partial x_j} \mathcal{D}u_i + \frac{\partial}{\partial x_i} \mathcal{D}u_j \right) + \delta_{ij} \frac{\mu}{1-2\mu} \sum_{k=1}^3 \frac{\partial}{\partial x_k} \mathcal{D}u_k \right] \left(\frac{\partial \delta u_i}{\partial x_j} + \frac{\partial \delta u_j}{\partial x_i} \right) d\Omega = \\ = \iiint_{\Omega} \sum_{i=1}^3 F_i \delta u_i d\Omega + \iint_{S_{\sigma}} \sum_{i=1}^3 t_i \delta u_i ds. \end{aligned} \quad (8)$$

Для численного определения перемещений u_1, u_2, u_3 на основании уравнения баланса (8) воспользуемся методом конечных элементов [10]. Перемещения можно определить через узловые перемещения (обобщенные координаты) q_p :

$$u_i = u_i(x_1, x_2, x_3, t) = \sum_{p=1}^M q_p(t) N_i^p(x_1, x_2, x_3), \quad i = 1, 2, 3, \quad (9)$$

где M – число степеней свободы; $p = 3(n-1) + j$; n – порядковый номер узла; j – номер координатного направления.

Тогда

$$\delta u_i = \sum_{p=1}^M \delta q_p N_i^p(x_1, x_2, x_3), \quad i = 1, 2, 3. \quad (10)$$

Подставим (9), (10) в (8), приведем подобные, вынесем из-под знака интеграла функции от времени $q_p(t)$. В результате получим

$$\begin{aligned} \frac{1}{2} \sum_{p=1}^M \mathcal{D}q_p \left\{ \iiint_{\Omega} \sum_{i=1}^3 \sum_{j=1}^3 \left[\frac{1}{2} \left(\frac{\partial N_i^p}{\partial x_j} + \frac{\partial N_j^p}{\partial x_i} \right) + \delta_{ij} \frac{\mu}{1-2\mu} \sum_{k=1}^3 \frac{\partial N_k^p}{\partial x_k} \right] \left[\sum_{r=1}^M \delta q_r \left(\frac{\partial N_i^r}{\partial x_j} + \frac{\partial N_j^r}{\partial x_i} \right) \right] d\Omega \right\} = \\ = \iiint_{\Omega} \left(\sum_{i=1}^3 F_i \left(\sum_{r=1}^M \delta q_r N_i^r \right) \right) d\Omega + \iint_{S_{\sigma}} \left(\sum_{i=1}^3 t_i \left(\sum_{r=1}^M \delta q_r N_i^r \right) \right) ds. \end{aligned} \quad (11)$$

После варьирования каждого из параметров δq_r , $r = 1, 2, \dots, M$ ($\delta q_r = 1$; $\delta q_i = 0$, $i \neq r$) из (11) получим систему обыкновенных дифференциальных уравнений для неизвестных значений $q_1, \dots, q_M$:

$$\begin{aligned} \frac{1}{2} \sum_{p=1}^M \mathcal{D}q_p \left\{ \iiint_{\Omega} \sum_{i=1}^3 \sum_{j=1}^3 \left[\frac{1}{2} \left(\frac{\partial N_i^p}{\partial x_j} + \frac{\partial N_j^p}{\partial x_i} \right) + \delta_{ij} \frac{\mu}{1-2\mu} \sum_{k=1}^3 \frac{\partial N_k^p}{\partial x_k} \right] \left(\frac{\partial N_i^r}{\partial x_j} + \frac{\partial N_j^r}{\partial x_i} \right) d\Omega \right\} = \\ = \iiint_{\Omega} \left(\sum_{i=1}^3 F_i N_i^r \right) d\Omega + \iint_{S_{\sigma}} \left(\sum_{i=1}^3 t_i N_i^r \right) ds, \quad r = 1, 2, \dots, M. \end{aligned} \quad (12)$$

Обозначим

$$\begin{aligned} A_{rp} = \iiint_{\Omega} \sum_{i=1}^3 \sum_{j=1}^3 \left[\frac{1}{2} \left(\frac{\partial N_i^p}{\partial x_j} + \frac{\partial N_j^p}{\partial x_i} \right) + \delta_{ij} \frac{\mu}{1-2\mu} \sum_{k=1}^3 \frac{\partial N_k^p}{\partial x_k} \right] \left(\frac{\partial N_i^r}{\partial x_j} + \frac{\partial N_j^r}{\partial x_i} \right) d\Omega, \\ Q_r(t) = \iiint_{\Omega} \left(\sum_{i=1}^3 F_i N_i^r \right) d\Omega + \iint_{S_{\sigma}} \left(\sum_{i=1}^3 t_i N_i^r \right) ds. \end{aligned}$$

Тогда (12) запишется в виде

$$\frac{1}{2} \sum_{p=1}^M A_{rp} \mathcal{D}q_p(t) = Q_r(t), \quad r = 1, 2, \dots, M,$$

или в развернутом виде

$$\sum_{p=1}^M A_{rp} (\eta \dot{q}_p(t) + G q_p(t)) = Q_r(t), \quad r = 1, 2, \dots, M. \quad (13)$$

Отметим, что уравнения системы (13) следует еще упростить (для узлов, соответствующих закрепленным участкам границы, следует соответствующие обобщенные координаты положить равными нулю).

Для системы уравнений (13) нужно также задать начальные условия

$$\mathbf{q}(0) = \mathbf{q}^0. \quad (14)$$

Их можно определить из начального условия

$$\mathbf{u}|_{t=0} = \mathbf{u}^0.$$

Для решения задачи Коши (13), (14) можно воспользоваться методами Рунге-Кутты с автоматическим управлением шагом интегрирования [5].

3. Построение математической модели слоя дорожной одежды как вязкоупругой среды (геометрически-нелинейный случай) и применение метода конечных элементов для ее численного анализа

Для геометрически-нелинейных задач компоненты тензора деформаций определяются следующими соотношениями Коши:

$$\varepsilon_{ij} = \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) + \sum_{d=1}^3 \frac{\partial u_d}{\partial x_i} \frac{\partial u_d}{\partial x_j}, \quad i, j = 1, 2, 3. \quad (15)$$

Для вариации деформации примем такое же выражение, как и в п. 2:

$$\delta \varepsilon_{ij} = \frac{1}{2} \left(\frac{\partial \delta u_i}{\partial x_j} + \frac{\partial \delta u_j}{\partial x_i} \right), \quad i, j = 1, 2, 3.$$

С учетом соотношений Коши (15) и (5) получим

$$\begin{aligned} \varepsilon_{cp.} &= \frac{1}{3} \sum_{k=1}^3 \frac{\partial u_k}{\partial x_k} + \frac{1}{3} \sum_{k=1}^3 \sum_{d=1}^3 \left(\frac{\partial u_d}{\partial x_k} \right)^2, \\ \sigma_{ij} &= \frac{1}{2} \left(\frac{\partial}{\partial x_j} \mathcal{D}u_i + \frac{\partial}{\partial x_i} \mathcal{D}u_j \right) + \delta_{ij} \frac{\mu}{1-2\mu} \sum_{k=1}^3 \frac{\partial}{\partial x_k} \mathcal{D}u_k + \\ &+ \sum_{d=1}^3 \mathcal{D} \left(\frac{\partial u_d}{\partial x_i} \frac{\partial u_d}{\partial x_j} \right) + \delta_{ij} \frac{\mu}{1-2\mu} \sum_{k=1}^3 \sum_{d=1}^3 \mathcal{D} \left(\frac{\partial u_d}{\partial x_k} \right)^2, \quad i, j = 1, 2, 3. \end{aligned} \quad (16)$$

Подставив (16) в (1), для неизвестных перемещений u_1, u_2, u_3 получим интегро-дифференциальное соотношение

$$\begin{aligned} &\frac{1}{2} \iiint_{\Omega} \sum_{i=1}^3 \sum_{j=1}^3 \left[\frac{1}{2} \left(\frac{\partial}{\partial x_j} \mathcal{D}u_i + \frac{\partial}{\partial x_i} \mathcal{D}u_j \right) + \delta_{ij} \frac{\mu}{1-2\mu} \sum_{k=1}^3 \frac{\partial}{\partial x_k} \mathcal{D}u_k \right] \left(\frac{\partial \delta u_i}{\partial x_j} + \frac{\partial \delta u_j}{\partial x_i} \right) d\Omega + \\ &+ \frac{1}{2} \iiint_{\Omega} \sum_{i=1}^3 \sum_{j=1}^3 \left[\sum_{d=1}^3 \mathcal{D} \left(\frac{\partial u_d}{\partial x_i} \frac{\partial u_d}{\partial x_j} \right) + \delta_{ij} \frac{\mu}{1-2\mu} \sum_{k=1}^3 \sum_{d=1}^3 \mathcal{D} \left(\frac{\partial u_d}{\partial x_k} \right)^2 \right] \left(\frac{\partial \delta u_i}{\partial x_j} + \frac{\partial \delta u_j}{\partial x_i} \right) d\Omega = \\ &= \iiint_{\Omega} \sum_{i=1}^3 F_i \delta u_i d\Omega + \iint_{S_{\sigma}} \sum_{i=1}^3 t_i \delta u_i ds. \end{aligned} \quad (17)$$

Для численного определения перемещений u_1, u_2, u_3 на основании уравнения баланса (17) также воспользуемся методом конечных элементов [10].

Перемещения будем искать в виде (9). Подставим (9), (10) в (17), приведем подобные, вынесем из-под знака интеграла функции от времени $q_p(t)$. В результате получим

$$\begin{aligned} & \frac{1}{2} \sum_{p=1}^M \mathcal{D}q_p \left\{ \iiint_{\Omega} \sum_{i=1}^3 \sum_{j=1}^3 \left[\frac{1}{2} \left(\frac{\partial N_i^p}{\partial x_j} + \frac{\partial N_j^p}{\partial x_i} \right) + \delta_{ij} \frac{\mu}{1-2\mu} \sum_{k=1}^3 \frac{\partial N_k^p}{\partial x_k} \right] \left[\sum_{r=1}^M \delta q_r \left(\frac{\partial N_i^r}{\partial x_j} + \frac{\partial N_j^r}{\partial x_i} \right) \right] d\Omega \right\} + \\ & + \frac{1}{2} \sum_{p=1}^M \mathcal{D}(q_p)^2 \iiint_{\Omega} \sum_{i=1}^3 \sum_{j=1}^3 \left[\sum_{d=1}^3 \left(\frac{\partial N_d^p}{\partial x_i} \frac{\partial N_d^p}{\partial x_j} + \delta_{ij} \frac{\mu}{1-2\mu} \sum_{k=1}^3 \left(\frac{\partial N_d^p}{\partial x_k} \right)^2 \right) \right] \left[\sum_{r=1}^M \delta q_r \left(\frac{\partial N_i^r}{\partial x_j} + \frac{\partial N_j^r}{\partial x_i} \right) \right] d\Omega + \\ & + \frac{1}{2} \sum_{p=1}^M \sum_{s=1, s \neq p}^M \mathcal{D}(q_p q_s) \iiint_{\Omega} \sum_{i=1}^3 \sum_{j=1}^3 \left[\sum_{d=1}^3 \left(\frac{\partial N_d^p}{\partial x_i} \frac{\partial N_d^s}{\partial x_j} + \frac{\partial N_d^p}{\partial x_j} \frac{\partial N_d^s}{\partial x_i} + \delta_{ij} \frac{2\mu}{1-2\mu} \sum_{k=1}^3 \left(\frac{\partial N_d^p}{\partial x_k} \frac{\partial N_d^s}{\partial x_k} \right) \right) \right] \times \\ & \times \left[\sum_{r=1}^M \delta q_r \left(\frac{\partial N_i^r}{\partial x_j} + \frac{\partial N_j^r}{\partial x_i} \right) \right] d\Omega = \iiint_{\Omega} \left(\sum_{i=1}^3 F_i \left(\sum_{r=1}^M \delta q_r N_i^r \right) \right) d\Omega + \iint_{S_{\sigma}} \left(\sum_{i=1}^3 t_i \left(\sum_{r=1}^M \delta q_r N_i^r \right) \right) ds. \quad (18) \end{aligned}$$

После варьирования каждого из параметров δq_r , $r=1, 2, \dots, M$ ($\delta q_r=1$; $\delta q_i=0$, $i \neq r$) из (18) получим нелинейную систему обыкновенных дифференциальных уравнений для неизвестных значений $q_1, \dots, q_M$:

$$\begin{aligned} & \frac{1}{2} \sum_{p=1}^M \mathcal{D}q_p \left\{ \iiint_{\Omega} \sum_{i=1}^3 \sum_{j=1}^3 \left[\frac{1}{2} \left(\frac{\partial N_i^p}{\partial x_j} + \frac{\partial N_j^p}{\partial x_i} \right) + \delta_{ij} \frac{\mu}{1-2\mu} \sum_{k=1}^3 \frac{\partial N_k^p}{\partial x_k} \right] \left(\frac{\partial N_i^r}{\partial x_j} + \frac{\partial N_j^r}{\partial x_i} \right) d\Omega \right\} + \\ & + \frac{1}{2} \sum_{p=1}^M \mathcal{D}(q_p)^2 \iiint_{\Omega} \sum_{i=1}^3 \sum_{j=1}^3 \left[\sum_{d=1}^3 \left(\frac{\partial N_d^p}{\partial x_i} \frac{\partial N_d^p}{\partial x_j} + \delta_{ij} \frac{\mu}{1-2\mu} \sum_{k=1}^3 \left(\frac{\partial N_d^p}{\partial x_k} \right)^2 \right) \right] \left(\frac{\partial N_i^r}{\partial x_j} + \frac{\partial N_j^r}{\partial x_i} \right) d\Omega + \\ & + \frac{1}{2} \sum_{p=1}^M \sum_{s=1, s \neq p}^M \mathcal{D}(q_p q_s) \iiint_{\Omega} \sum_{i=1}^3 \sum_{j=1}^3 \left[\sum_{d=1}^3 \left(\frac{\partial N_d^p}{\partial x_i} \frac{\partial N_d^s}{\partial x_j} + \frac{\partial N_d^p}{\partial x_j} \frac{\partial N_d^s}{\partial x_i} + \delta_{ij} \frac{2\mu}{1-2\mu} \sum_{k=1}^3 \left(\frac{\partial N_d^p}{\partial x_k} \frac{\partial N_d^s}{\partial x_k} \right) \right) \right] \left(\frac{\partial N_i^r}{\partial x_j} + \frac{\partial N_j^r}{\partial x_i} \right) d\Omega = \\ & = \iiint_{\Omega} \left(\sum_{i=1}^3 F_i N_i^r \right) d\Omega + \iint_{S_{\sigma}} \left(\sum_{i=1}^3 t_i N_i^r \right) ds, \quad r=1, 2, \dots, M. \quad (19) \end{aligned}$$

Обозначим

$$\begin{aligned} B_{rp} &= \iiint_{\Omega} \sum_{i=1}^3 \sum_{j=1}^3 \left[\sum_{d=1}^3 \left(\frac{\partial N_d^p}{\partial x_i} \frac{\partial N_d^p}{\partial x_j} + \delta_{ij} \frac{\mu}{1-2\mu} \sum_{k=1}^3 \left(\frac{\partial N_d^p}{\partial x_k} \right)^2 \right) \right] \left(\frac{\partial N_i^r}{\partial x_j} + \frac{\partial N_j^r}{\partial x_i} \right) d\Omega, \\ C_{rps} &= \iiint_{\Omega} \sum_{i=1}^3 \sum_{j=1}^3 \left[\sum_{d=1}^3 \left(\frac{\partial N_d^p}{\partial x_i} \frac{\partial N_d^s}{\partial x_j} + \frac{\partial N_d^p}{\partial x_j} \frac{\partial N_d^s}{\partial x_i} + \delta_{ij} \frac{2\mu}{1-2\mu} \sum_{k=1}^3 \left(\frac{\partial N_d^p}{\partial x_k} \frac{\partial N_d^s}{\partial x_k} \right) \right) \right] \left(\frac{\partial N_i^r}{\partial x_j} + \frac{\partial N_j^r}{\partial x_i} \right) d\Omega. \end{aligned}$$

Тогда (19) запишется в виде

$$\frac{1}{2} \sum_{p=1}^M A_{rp} \mathcal{D}q_p(t) + \frac{1}{2} \sum_{p=1}^M B_{rp} \mathcal{D}(q_p(t))^2 + \frac{1}{2} \sum_{p=1}^M \sum_{s=1, s \neq p}^M C_{rps} \mathcal{D}(q_p(t) q_s(t)) = Q_r(t), \quad r=1, 2, \dots, M,$$

или в развернутом виде

$$\sum_{p=1}^M A_{rp}(\eta \dot{q}_p(t) + G q_p(t)) + \sum_{p=1}^M B_{rp} q_p(2\eta \dot{q}_p(t) + G q_p(t)) + \\ + \sum_{p=1}^M \sum_{\substack{s=1 \\ s \neq p}}^M C_{rps}(\eta q_s(t) \dot{q}_p(t) + \eta q_p(t) \dot{q}_s(t) + G q_p(t) q_s(t)) = Q_r(t), \quad r = 1, 2, \dots, M. \quad (20)$$

Отметим, что, также как и в линейном случае, уравнения системы (20) следует еще упростить (для узлов, соответствующих закрепленным участкам границы, необходимо соответствующие обобщенные координаты положить равными нулю).

Для системы уравнений (20) нужно также задать начальные условия

$$\mathbf{q}(0) = \mathbf{q}^0. \quad (21)$$

Их можно определить из начального условия

$$\mathbf{u}|_{t=0} = \mathbf{u}^0.$$

Для решения задачи Коши (20), (21) можно воспользоваться методами Рунге-Кутты с автоматическим управлением шагом интегрирования [5].

Выводы

Таким образом, впервые для математического описания напряженно-деформированного состояния слоя дорожной одежды как вязкоупругой среды с линейными или геометрически-нелинейными свойствами на основании одноэлементной модели Кельвина построены соответствующие интегральные тождества. Для их численного анализа описано применение метода конечных элементов: построены задачи Коши для узловых перемещений. Полученные результаты могут быть использованы для создания программного продукта при расчете напряженно-деформированного состояния многослойных нежестких дорожных одежд, а также для методических рекомендаций по определению расчетных вязких и упругих характеристик асфальтобетонов. Кроме того, предлагаемая методика позволяет моделировать реакцию слоев дорожных одежд с учетом вязкоупругих свойств материалов, что дает более реалистичную картину состояния покрытия при нагрузках. Этим и определяется *научная новизна и практическая значимость* полученных результатов.

Список литературы: 1. *Автомобільні дороги*. Визначення транспортно-експлуатаційних показників дорожніх одягів: СОУ 45.2-000 18112-042: 2009. Вид. офіц. К.: Укравтодор: Стандарт Укравтодор, 2009. 46 с. 2. *Асфальтобетон дорожній. Метод випробування на стійкість до накопичення залишкових деформацій*: СОУ 45.2-00018112-020; 2009. Офіц. вид. К.: Укравтодор, 2009. 11 с. 3. *Асфальтобетон дорожній. Методика оцінки зчеплення між асфальтобетонними шарами при зсуві*: СОУ 45.2 – 00018112 – 046: 2009. Офіц. вид. К.: Державна служба автомобільних доріг України (Укравтодор) 2009. 11 с. 4. *Бленд Д.* Теория линейной вязкоупругости. М.: Мир, 1965. 199 с. 5. *Вержбицкий В.М.* Основы численных методов. М.: Высш. шк., 2002. 840 с. 6. *Инструкция по проектированию дорожных одежд нежесткого типа*: ВСН 46 – 83 – Минтрансстрой. М.: Транспорт, 1985. 157 с. 7. *Мейз Дж.* Теория и задачи механики сплошных сред. М.: Мир, 1974. 318 с. 8. *Lemaitre J., Chaboche J.* Mechanics of solid materials. Cambridge University Press, 1990. 556 p. 9. *Marvalova B.* Viscoelastic properties of filled rubber. Experimental observations and material modeling // Engineering Mechanics. 2007. Vol. 14, No. 1/2. P. 81 – 89. 10. *Zienkiewicz O.C., Taylor R.L.* The Finite Element Method. Vol. 1: The Basics. Oxford: BH, 2000. 689 p. 11. *Zienkiewicz O.C., Taylor R.L.* The Finite Element Method. Vol. 2: Solid Mechsnics. Oxford: BH, 2000. 459 p.

Поступила в редколлегию 13.12.2012

Богомолов Виктор Александрович, д-р техн. наук, проф., заместитель ректора (проректор) ХНАДУ. Научные интересы: математическое моделирование и методы численного анализа физико-механических полей. Увлечения и хобби: волейбол. Адрес: Украина, 61002, ул. Петровского, 25, тел. (057) 7003863.

Иевлева Светлана Николаевна, канд. техн. наук, доц. прикладной математики ХНУРЭ. Научные интересы: математическое моделирование сложных систем. Увлечения и хобби: нумизматика. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. (057) 7021436.

Разницын Илья Львович, канд. физ.-мат. наук, доц., с.н.с. каф. автомобилей ХНАДУ. Научные интересы: математическое моделирование физико-механических полей, краевые задачи для дифференциальных уравнений в частных производных. Адрес: Украина, 61002, ул. Петровского, 25, тел. (057) 7073737.

Сидоров Максим Викторович, канд. физ.-мат. наук, доцент каф. прикладной математики ХНУРЭ. Научные интересы: математическое моделирование, численные методы, математическая физика, теория R-функций и её приложения, стохастический анализ и его приложения. Увлечения и хобби: всемирная история, история искусств. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. (057) 7021436.

ПОСТРОЕНИЕ ЛИНЕАРИЗИРОВАННЫХ МАТЕМАТИЧЕСКИХ МОДЕЛЕЙ РАБОТЫ ВАКУУМ-АППАРАТОВ КРИСТАЛЛИЗАЦИОННОГО ОТДЕЛЕНИЯ САХАРНОГО ЗАВОДА

Рассматривается работа отделения кристаллизации сахарного завода. Предлагаются линеаризованные математические модели работы вакуум-аппаратов, необходимые для использования в автоматизированных системах управления технологическими процессами кристаллизационного отделения.

1. Введение

Развитие производства сахарной продукции в Украине – одно из основных стратегических направлений в сельскохозяйственном производстве. Главным производственным процессом получения сахара на сахарном заводе является процесс кристаллизации. Кристаллизация сахара – завершающий этап в его производстве. Здесь выделяют практически чистую сахарозу из многокомпонентной смеси, которой является сироп. В сокоочистительном отделении из диффузионного сока удаляется около 1/3 несахаров, остальные несахара вместе с сахарозой поступают в продуктовое (кристаллизационное) отделение, где большая часть сахарозы выкристаллизовывается в виде сахара-песка, а несахара остаются в межкристальном растворе. Выход сахара на 75% зависит от потерь сахара в мелассе. Потери в продуктовом отделении определяют технико-экономические показатели завода. Качество сахара прямо связано с потерями его в мелассе. Задачей оптимизации технологического процесса является выбор между глубоким истощением мелассы и качеством песка. Задача получения сахара стандартного качества решается с помощью технологии многоступенчатой кристаллизации раствора, позволяющей свести потери к минимуму. Наиболее распространены двухступенчатая и трехступенчатая схемы работы продуктового отделения. Для получения сахара хорошего качества необходимо использовать гибкие схемы производства, предусматривающие оперативное перераспределение потоков и режимов работы в соответствии с ситуацией на заводе и качеством перерабатываемого сырья [1].

Рациональная технологическая схема продуктового отделения должна иметь столько ступеней кристаллизации, чтобы ее суммарный эффект составлял 30-33%, а коэффициент завода составлял бы более 90% при среднем качестве свеклы. В достоинства трехпродуктовой схемы работы можно включить более высокий выход продукции (37%) и высокое качество получаемого товарного продукта. От прочих схем кристаллизации она отличается прямоточностью, в которой существует один рециркуляционный контур - возврат клеровки. Исходным сырьем для продуктового отделения является сульфитированная смесь сиропа с клеровкой сахаров II кристаллизации и сахара-аффинада III кристаллизации с чистотой не менее 92%.

В сахарном производстве технологический процесс получения сахара постоянно меняется в зависимости от изменения качества продукции.

Исходя из изложенного, можно сделать заключение, что процесс кристаллизации, как и все другие основные процессы переработки сырья в заводе, является сложным динамическим процессом. Для получения оптимального выхода продукции необходимо уделить внимание вопросам внедрения современных автоматизированных систем управления технологическими процессами (АСУТП).

Эффективный выход продукции производства зависит от системы управления отделением кристаллизации на базе программно-технических комплексов (ПТК), обеспечивающих управление технологическим процессом в отделении кристаллизации по заданным безопасным технологическим и экономическим критериям, которые определяют качественные и количественные показатели выхода сахара и сопутствующей продукции отделения. Современная АСУТП включает в себя технические средства, программное обеспечение и

оператора, роль которого в автоматизированном процессе производства сводится к наладке, регулировке и обслуживанию средств автоматизации наблюдению за их действием.

АСУТП отделения кристаллизации представляет собой распределённую систему управления малого масштаба, как и в диффузионном, сокоочистительном и выпарном отделениях, включающую подсистемы сбора и отображения информации, автоматического регулирования, дискретно-логического управления, противоаварийных защит и блокировок. Она предоставляет оператору подробную информацию о протекании технологического процесса, производит архивирование основных технологических параметров, ведёт протокол событий, позволяет формировать отчёты в различных видах и имеет возможность самодиагностики. Эти свойства повышают надёжность и удобство эксплуатации системы, а также повышают безопасность эксплуатации оборудования.

При помощи АСУТП в отделении кристаллизации достигается улучшение: режима уваривания сиропа до пресыщения в вакуум – аппаратах под разрежением, зарождение и наращивание кристаллов и последующее разделение в центрифугах на сахар-песок и первый оттек.

Важную роль в системе управления технологическим процессом кристаллизации играет алгоритм управления, представляющий собой описание процедуры обработки информации о наблюдаемых переменных состояния в целях определения управляющих воздействий, реализуемых для получения требуемых показателей управляемого процесса как в установившемся, так и в переходном режиме.

Алгоритм управления должен четко определять последовательность действий, необходимых для того, чтобы наилучшим образом решить поставленную задачу. Он составляется технологом для упрощения задачи программиста при написании программного обеспечения для контроллера.

Исходным этапом построения модели является расчет и анализ статики процесса, т.е. рассмотрение данных о равновесии, на основе которых определяют направление протекания и возможные пределы осуществления процесса. Исходя из знания значений и параметров технологического процесса получения сахара (на всех этапах производства (диффузия, дефекация, выпарка, кристаллизация)), на основании законов сохранения массы и энергии составляются уравнения материального и энергетического баланса. Зная эти уравнения, можно определить кинетику процесса и скорость его протекания при различных изменениях.

2. Основной материал

Для оптимальной работы всего кристаллизационного отделения следует соблюдать температурный режим во всех вакуум-аппаратах отделения. Составим уравнение теплового баланса процесса кристаллизации в вакуум-аппаратах [2]:

$$Q_{\text{пар}} + Q_{\text{кр}}^i = Q_{\text{нагр}}^{ij} + Q_{\text{выпар}}^j, \quad (1)$$

где $Q_{\text{пар}}$ – количество тепла, которое поступает с паром; $Q_{\text{кр}}$ – количество тепла, получаемое при кристаллизации в i -м аппарате; $Q_{\text{нагр}}^{ij}$ – количество тепла, затраченное на нагрев продукта j -го аппарата; $Q_{\text{выпар}}$ – количество тепла, потраченное на выпаривание в j -м аппарате.

Каждую из составляющих уравнения (1) можно представить в виде:

$$Q_{\text{пар}} = G_{\text{пар}} C_{\text{пар}} \theta_{\text{пар}}, \quad (2)$$

здесь $G_{\text{пар}}$ – количество пара; $C_{\text{пар}}$ – теплоемкость пара; $\theta_{\text{пар}}$ – температура пара;

$$Q_{\text{кр}} = G_{\text{кр}} C_{\text{кр}} \theta_{\text{кр}}, \quad (3)$$

где $G_{\text{кр}}$ – количество продукта кристаллизации; $C_{\text{кр}}$ – теплоемкость продукта кристаллизации; $\theta_{\text{кр}}$ – температура продукта кристаллизации;

$$Q_{\text{нагр}} = G_{\text{нагр}} C_{\text{нагр}} \theta_{\text{нагр}}, \quad (4)$$

$G_{\text{нагр}}$ – количество исходного продукта; $C_{\text{нагр}}$ – теплоемкость исходного продукта, $\theta_{\text{нагр}}$ – температура исходного продукта;

$$Q_{\text{выпар}} = G_{\text{выпар}} C_{\text{выпар}} \theta_{\text{выпар}}, \quad (5)$$

где $G_{\text{выпар}}$ – количество выпаренного растворителя, $C_{\text{выпар}}$ – теплоемкость выпаренного растворителя; $\theta_{\text{выпар}}$ – температура выпаренного растворителя.

Соответственно получим уравнение теплового баланса для вакуум-аппарата первой кристаллизации:

$$G_{\text{пар}} C_{\text{пар}} \theta_{\text{пар}} + G_{\text{кр}} C_{\text{кр}} \theta_{\text{кр}} = G_{\text{нагр}} C_{\text{нагр}} \theta_{\text{нагр}} + G_{\text{выпар}} C_{\text{выпар}} \theta_{\text{выпар}}. \quad (6)$$

При нарушении теплового баланса уравнение (6) примет вид:

$$V_1 \rho c_1 \frac{d\theta_{\text{нагр}}}{d\tau} = \Delta (G_{\text{пар}} C_{\text{пар}} \theta_{\text{пар}} + G_{\text{кр}} C_{\text{кр}} \theta_{\text{кр}} - G_{\text{нагр}} C_{\text{нагр}} \theta_{\text{нагр}} - G_{\text{выпар}} C_{\text{выпар}} \theta_{\text{выпар}}), \quad (7)$$

где V_1 – объем первого вакуум-аппарата; ρ – плотность сиропа; c_1 – удельная теплоемкость вещества в первом аппарате.

Отклонения переменных в кристаллизаторе, с учетом линеаризации, примут следующий вид:

$$\frac{d\theta_{\text{нагр}}^{01}}{d\tau} = \frac{1}{V_1 \rho c_1} (G_{\text{пар}} C_{\text{пар}} \Delta \theta_{\text{пар}} + C_{\text{пар}} \theta_{\text{пар}} \Delta G_{\text{пар}} + G_{\text{кр}}^1 C_{\text{кр}}^1 \Delta \theta_{\text{кр}}^1 + C_{\text{кр}}^1 \theta_{\text{кр}}^1 \Delta G_{\text{кр}}^1 - G_{\text{нагр}}^{01} C_{\text{нагр}}^{01} \Delta \theta_{\text{нагр}}^{01} - C_{\text{нагр}}^{01} \theta_{\text{нагр}}^{01} \Delta G_{\text{нагр}}^{01} - G_{\text{выпар}}^1 C_{\text{выпар}}^1 \Delta \theta_{\text{выпар}}^1 - C_{\text{выпар}}^1 \theta_{\text{выпар}}^1 \Delta G_{\text{выпар}}^1). \quad (8)$$

Соответственно для остальных вакуум-аппаратов уравнения составляются аналогично:

$$\frac{d\theta_{\text{нагр}}^{12}}{d\tau} = \frac{1}{V_2 \rho c_2} (G_{\text{пар}} C_{\text{пар}} \Delta \theta_{\text{пар}} + C_{\text{пар}} \theta_{\text{пар}} \Delta G_{\text{пар}} + G_{\text{кр}}^2 C_{\text{кр}}^2 \Delta \theta_{\text{кр}}^2 + C_{\text{кр}}^2 \theta_{\text{кр}}^2 \Delta G_{\text{кр}}^2 - G_{\text{нагр}}^{12} C_{\text{нагр}}^{12} \Delta \theta_{\text{нагр}}^{12} - C_{\text{нагр}}^{12} \theta_{\text{нагр}}^{12} \Delta G_{\text{нагр}}^{12} - G_{\text{выпар}}^2 C_{\text{выпар}}^2 \Delta \theta_{\text{выпар}}^2 - C_{\text{выпар}}^2 \theta_{\text{выпар}}^2 \Delta G_{\text{выпар}}^2), \quad (9)$$

$$\frac{d\theta_{\text{нагр}}^{23}}{d\tau} = \frac{1}{V_3 \rho c_3} (G_{\text{пар}} C_{\text{пар}} \Delta \theta_{\text{пар}} + C_{\text{пар}} \theta_{\text{пар}} \Delta G_{\text{пар}} + G_{\text{кр}}^3 C_{\text{кр}}^3 \Delta \theta_{\text{кр}}^3 + C_{\text{кр}}^3 \theta_{\text{кр}}^3 \Delta G_{\text{кр}}^3 - G_{\text{нагр}}^{23} C_{\text{нагр}}^{23} \Delta \theta_{\text{нагр}}^{23} - C_{\text{нагр}}^{23} \theta_{\text{нагр}}^{23} \Delta G_{\text{нагр}}^{23} - G_{\text{выпар}}^3 C_{\text{выпар}}^3 \Delta \theta_{\text{выпар}}^3 - C_{\text{выпар}}^3 \theta_{\text{выпар}}^3 \Delta G_{\text{выпар}}^3), \quad (10)$$

$$\frac{d\theta_{\text{нагр}}^{34}}{d\tau} = \frac{1}{V_4 \rho c_4} (G_{\text{пар}} C_{\text{пар}} \Delta \theta_{\text{пар}} + C_{\text{пар}} \theta_{\text{пар}} \Delta G_{\text{пар}} + G_{\text{кр}}^4 C_{\text{кр}}^4 \Delta \theta_{\text{кр}}^4 + C_{\text{кр}}^4 \theta_{\text{кр}}^4 \Delta G_{\text{кр}}^4 - G_{\text{нагр}}^{34} C_{\text{нагр}}^{34} \Delta \theta_{\text{нагр}}^{34} - C_{\text{нагр}}^{34} \theta_{\text{нагр}}^{34} \Delta G_{\text{нагр}}^{34} - G_{\text{выпар}}^4 C_{\text{выпар}}^4 \Delta \theta_{\text{выпар}}^4 - C_{\text{выпар}}^4 \theta_{\text{выпар}}^4 \Delta G_{\text{выпар}}^4). \quad (11)$$

На основе полученных соотношений могут быть реализованы эффективные подходы для синтеза системы управления.

3. Выводы

Исследовав работу отделения кристаллизации сахарного производства, можно сделать выводы, что одними из основных устройств, отвечающих за качество и количество сахара, являются вакуум-аппараты периодического действия. Нормальная работа данных аппаратов заключается в поддержании оптимальных значений количества сиропа при заданных температурах, для устранения остатков влаги из сиропа и получения кристаллов сахара. Были построены уравнения теплового баланса для всех вакуум-аппаратов отделения, из данных уравнений получены линеаризованные математические модели работы теплообменных частей кристаллизаторов. Учитывая линейность соотношений, возможно эффективно использовать при построении систем управления.

Список литературы: 1. Ляшенко С.А., Ляшенко А.С., Беляева И.С. Концепции повышения эффективности АСУ ТП при производстве сахара в Украине // Вісник Харківського національного технічного університету сільського господарства імені Петра Василенка. “Сучасні напрямки технології та механізації процесів переробних і харчових виробництв”. Харків. 2008. Вип. 74. С. 54-63. 2. Дідур В.А., Стручаєв М.І. Теплотехніка, теплопостачання і використання теплоти в сільському господарстві / За заг. ред. В.А. Дідура. К.: Аграрна освіта, 2008. 233с.

Поступила в редколлегию 11.12.2012

Ляшенко Сергей Алексеевич, канд. техн. наук, доцент кафедры БЖ ХНТУСХ им. П.Василенко. Научные интересы: моделирование сложных систем управления. Адрес: Украина, 61002, Харьков, ул. Артема, 44, тел. 732-76-26.

Ляшенко Алексей Сергеевич, канд. техн. наук, доцент кафедры ЭВМ ХНУРЭ. Научные интересы: автоматизированные системы управления. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 702-13-54.

ОБ ОДНОМ ПОДХОДЕ К МОДУЛЬНОМУ ПРОЕКТИРОВАНИЮ ИНФОРМАЦИОННЫХ СИСТЕМ

Предлагается подход к проектированию информационных систем на основе структурно-логического синтеза универсальных модулей. Каждый модуль соответствует одной типовой функциональной компоненте при проектировании информационной системы: от модуля, реализующего тип модели данных, до типовых модулей построения интерфейса пользователя. В качестве интегральной оболочки синтезируемой информационной системы предлагается фреймовая модель.

Введение

Разработка и реализация современных крупных информационных проектов имеет, как правило, затяжной характер, их стоимость превосходит запланированную, а окончательный продукт получается ненадежным и сложным в сопровождении [1]. Это привело к ситуации, которая известна под названием «кризис программного обеспечения». Хотя первые упоминания о кризисе были сделаны еще в конце 80-х годов, даже спустя 30 лет его все еще не удалось преодолеть. В настоящее время многие авторы видят причины этого кризиса в следующем:

- разработка около 40% систем заканчивается неудачно или прекращается до завершения работ;
- рационально интегрировать интересы бизнеса и используемой информационной технологии удается не более чем в 25% проектируемых систем;
- только 20-30% информационных систем отвечают всем критериям достижения успеха.

Основные неудачи при создании программного обеспечения вызваны отсутствием полной спецификации всех требований на этапе проектирования, приемлемой методологии разработки или недостаточной степенью разделения общего глобального проекта на отдельные компоненты, поддающиеся эффективному контролю и управлению. Для решения этих проблем в практике проектирования информационных систем был предложен структурный подход, называемый жизненным циклом разработки программного обеспечения.

Под жизненным циклом разработки информационных систем традиционно понимается упорядоченная совокупность этапов, обеспечивающих создание качественного программного продукта. В литературе существует множество нареканий на несовершенство данного структурирования, но смысл его в целом достаточно ясен – борьба со сложностью процесса разработки программных продуктов путем разделения этапов и локализации только тех задач, которые могут и должны решаться именно здесь [2].

Обзор современных методов проектирования информационных систем

Технология программирования уже прошла достаточно долгий путь развития, и сейчас происходит переоценка ее фундаментальных исходных посылок. Первым кандидатом для такой переоценки стала традиционная точка зрения на процесс разработки как на процесс, основанный на понятии жизненного цикла [3].

Существуют различные подходы к разработке информационных моделей и систем. Одно из таких направлений представляет W/O (Warnier/Orr) методология DSSD (Data Structured System Development), которая объединяет методологию Warnier по использованию логических структур данных и логических конструкций программ. Такая методология предполагает, что в распоряжении проектировщика информационной системы имеется представительный набор процедурных шаблонов для широкого класса программируемых задач. В настоящее время DSSD-методология «переросла» из методологии разработки программ в методологию разработки систем.

Следующим подходом к созданию моделей программ является логическое моделирование Гэйна. Логическая модель системы проектируется в процессе последовательного применения таких этапов:

- описание природы предметной области с помощью диаграмм потоков данных (Data Flow Diagrams);
- сведение полученной на предыдущем этапе информации в двумерные таблицы, которые в дальнейшем нормализуются;
- коррекция DFD с учетом результатов нормализации предыдущего этапа;
- разбиение полученной в результате выполнения предыдущих этапов модели на «процедурные единицы», а также определение деталей каждой процедурной единицы.

Третьим в ряду подходов к созданию модели проектируемой программы является метод Иордана. Он включает два компонента: инструментальные средства и методики. Методология Иордана ориентирована на проектирование систем обработки данных. Под инструментальными средствами здесь понимаются различные диаграммы, используемые при описании моделей требований и моделей архитектуры проектируемой информационной системы. Самые известные из таких диаграмм - диаграммы потоков данных DFD. Однако их недостатком является отсутствие средств описания отношений между данными и их «поведения» во времени. Вот почему в инструментальные средства метода Иордана на сегодняшний день кроме DFD включены ERD-диаграммы (Entity Relationship Diagrams) и STD-диаграммы (State Transition Diagrams).

Подход, основанный на методе Иордана, помогает перейти от бланка на бумаге и/или экранной формы к хорошо организованной системной модели. Первоначально эти методики базировались на традиционном top-down проектировании. В настоящее время используется метод событийного разбиения (event partitioning). При этом сначала создается контекстная диаграмма верхнего уровня, где определяются системные ограничения и интерфейсы с «внешним миром». Затем с помощью техники интервью формируется список событий из внешней среды, на которые система должна реагировать. Такой подход обеспечивает простой базис для формирования «сырой» DFD. Несколько DFD-реакций могут быть объединены в редакцию более высокого уровня. Проблемы, возникающие при использовании метода ERD-диаграмм, связаны, прежде всего, с трудностями интеграции компонентов при разработке всей системы. Вот почему в последнее время этот метод был обобщен за счет введения интегрированной базы данных. При этом ERD-метод трансформируется в структурную методологию, основные этапы которой сводятся к разработке ERD [4].

Последним методом, который рассматривается в данном обзоре, является метод структурного проектирования. Структурное проектирование сделалось действительно мощным и активно используемым на практике подходом из-за того, что к моделям и методам были добавлены оценки результатов проектирования. Здесь предлагаются все проектные решения располагать в трехмерном пространстве «содержание – сложность – связность». И утверждается, что хорошими проектными решениями будут лишь те, которые при заданном содержании имеют минимальную сложность и максимальную связность.

Постановка задачи

В связи с изложенными выше тенденциями в проектировании информационных систем предлагается на основании анализа реляционных моделей данных различных прикладных информационных систем разработать типовые структурно-логические модули, соответствующие классам задач проектирования реляционных моделей данных, а также типовые интерфейсы пользователя. Каждый такой модуль включает функционально полную реляционную модель для решения задачи хранения и обработки данных на основе реляционной СУБД (системы управления базами данных) и интерфейс пользователя.

Решение поставленной задачи

В качестве структурно-логической единицы, соответствующей синтезируемой информационной системе, предлагается фреймовая модель. Фрейм – каркас, рамка, который объединяет в себе слоты – необходимые компоненты информационной системы. В общем виде такая модель может быть записана следующим образом:

$$FR\{< R_1, C_{11}, C_{12}, \dots, C_{1m}, >, \dots, < R_2, C_{21}, C_{22}, \dots, C_{2m}, >, \dots, < R_{k1}, C_{km}, >\}, \quad (1)$$

где FR – имя фрейма (идентификатор проектируемой информационной системы); пара $\langle Ri, Ci \rangle$ – i -й слот фрейма; Ri – имя слота; Ci – значение слота.

Слот в модели (1) является структурно-логической конструкцией для реализации конкретных функций информационной системы. Слоты из фрейма можно удалять, добавлять, изменять их функциональное назначение. Для удобства дальнейших рассуждений слот-компонент фрейма определим как e . Любому компоненту-слоту e будет соответствовать определенная арность $\text{Dim}(e)$, которая показывает число модулей сопряжения. Модули сопряжения компонент-слотов могут быть описаны набором признаков или параметров.

Таким образом, одним из факторов, определяющих функциональные возможности компонент-слотов каждого класса, является наличие и типы встроенных модулей сопряжения. Другим фактором является различная программная реализация компонент-слотов и тип потребляемого в режиме функционирования ресурса. Конкретной программной реализации компонента-слота e функциональной элементной базы соответствует его представление вектором параметров $V(e)$.

Любой компонент-слот e функциональной базы можно представить следующими образом $e = [I(e), G(e), V(e)]$, где $I(e)$ – номер класса компонента-слота; $G(e)$ – тип структуры модулей сопряжения; $V(e)$ – вектор конструктивных параметров.

Для формального определения возможных функциональных вариантов компонент-слотов введем преобразование подобия s , отображающее множество E на себя $s: E \rightarrow E$. Два образующих компонента-слота e_1 и e_2 подобны, если существует некоторое преобразование подобия $s \in S$, такое, что $se_1 = e_2$. Множество S содержит среди прочих преобразование подобия вида s_0 , для которого $s_0 e = e$. Конкретный вид преобразований подобия зависит от проектируемого семейства информационных систем.

Будем считать, что используемые преобразования подобия удовлетворяют следующим требованиям:

- множество S преобразований подобия S является группой;
- любое $s \in S$ отображает класс E_i в себя при любом номере класса компонент-слотов;
- элементы множества S не влияют на структуру модулей сопряжения, но могут влиять на их признаки.

Таким образом, структура модулей сопряжения и индекс класса компонентов инвариантны относительно преобразования подобия.

С учетом свойства инвариантности для компонента-слота можно ввести понятие функционального класса компонента-слота: $z = [I(e), G(e)]$, которое отражает наиболее важную информацию о компонентах-слотах, заключенную в номере класса и структуре модулей сопряжения. Для функциональной базы E можно определить интегральную категорию верхнего уровня – функциональную базу типов. Рассмотрим отображение $h: E \rightarrow E'$, где E и E' – две функциональные базы одного и того же семейства, связанные с одной и той же группой преобразований подобия S .

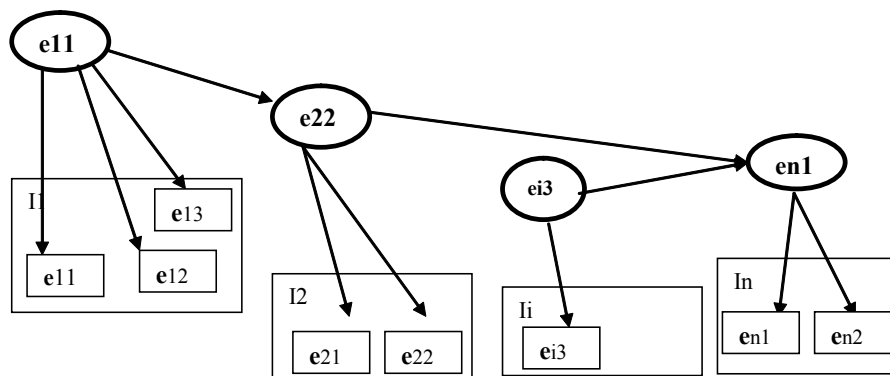
Будем называть преобразование h инвариантом связей, если для любого компонента-слота $e \in E$ компоненты e и $e' = h(e)$ имеют одни и те же программные модули сопряжения, а также $se \rightarrow se'$ для $\forall s \in S, \forall e \in E$. С помощью отображения h можно осуществить переход от функциональной базы E к новой элементной базе E' , не нарушая общей структуры информационных систем, в которых используются компоненты E .

Данный случай возникает при замене имеющейся функциональной базы семейства элементной базой нового поколения. Структура элементной базы семейства формируется классами компонент-слотов. Внутри класса слоты отличаются структурой модулей

сопряжения $G(e)$. Компоненты одного класса, отличающиеся какими-либо из перечисленных выше характеристик и признаков, будем называть модификациями компонент-слотов класса. Модификации компонентов учитываем различными номерами компонент-слотов; так, при двухсимвольном обозначении e_{ij} первый символ указывает класс компонента-слота, второй – номер модификации.

Компоненты класса, связанные с другими его компонентами одним и тем же преобразованием подобия S , являются универсальными. Число универсальных компонент-слотов в классе определяется числом используемых преобразований подобия.

На рисунке представлена структурная схема унифицированной модульной системы синтеза информационной системы на основе структурно-логических элементов. **I1** – блок унифицированных типовых реляционных схем баз данных; **I2** – блок унифицированных типовых интерфейсов пользователя, который состоит из модулей: **e21** – «главная кнопочная форма», **e22** – «ленточная форма» и т.д. **In** – блок унифицированных типовых выходных форм пользователя.



Структура универсальных модулей для синтеза информационных систем

Выводы

Предложен подход к синтезу типовых информационных систем основанный на унифицированных структурно-логических модулях, соответствующих классам задач проектирования реляционных моделей данных, а также типовых интерфейсах пользователя. Каждый такой модуль включает функционально полную реляционную модель, для решения задачи хранения и обработки данных на основе реляционной системы управления базами данных и интерфейс пользователя. Предлагаемый подход позволит значительно увеличить эффективность процесса проектирования информационных систем на основе реляционных баз данных.

Дальнейшие исследования предполагается проводить в направлении использования предложенных структурно-логических модулей для построения автоматизированных алгоритмов синтеза информационных систем.

Список литературы: 1. Пасічник В. В. Організація баз даних та знань / В. В. Пасічник, В. А. Резніченко. К. : БНУ, 2006. 386 с. 2. Филатов В. А. Методы и средства проектирования информационных систем и распределенных баз данных / В. А. Филатов, Р. В. Семенец // Вестник Херсонского национального технического университета. 2007. № 4(27). С. 203-207. 3. Филатов В. А. Концепция проектирования средств информационной поддержки в системе государственного управления / В. А. Филатов, Е. И. Фалькович // Збірник наукових праць Української Академії державного управління при Президенті України. Харків. 2001. Вип. 1. С. 68-70. 4. Йордан Э. Структурное программирование и конструирование программ: Пер. с англ. М.: Мир, 1979. 415 с.

Поступила в редколлегию 11.10.2012

Божинский Иван Андреевич, канд. техн. наук, зам. начальника НИЧ ХНУРЭ. Научные интересы: информационные системы. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-378.

ПОВЫШЕНИЕ ЭФФЕКТИВНОСТИ ИММУННЫХ МЕТОДОВ КЛАССИФИКАЦИИ ОБЪЕКТОВ С ПОМОЩЬЮ ЦЕЛЕВОГО КЛОНАЛЬНОГО ОТБОРА

Рассматривается иммунный алгоритм классификации объектов, работающий на основе модели отбора клонов при использовании последовательного клонирования с конкурентно-целевым отбором. Основное внимание уделяется операторам мутации и первичного отбора, принципам отбора целевых антигенов. Для классификации антител, не прошедших первичный отбор, предлагается использование антител-делегатов с известной принадлежностью к классам.

1. Введение

В настоящее время в сфере моделирования искусственного интеллекта большое внимание уделяется системам обработки информации, функционирующим на основе биологических принципов организации вычислений. В связи с этим большим исследовательским интересом пользуются модели искусственных иммунных систем (ИИС) и их практические реализации. Различные иммунные модели и методы успешно применяются для решения задач классификации и кластеризации данных [1, 2]. Процесс классификации объектов на основе ИИС разделяется на два этапа: на первом этапе происходит восстановление признаков классов исходными объектами с помощью иммунных операторов клонирования и отбора клонов, на втором этапе производится определение классов для объектов, не отнесенных ни к одному классу.

Наиболее часто используемыми иммунными методами классификации являются методы, функционирующие на основе модели отбора клонов, и методы на основе модели иммунной сети [1, 2]. При применении метода CLONALG [1, 2] для классификации объектов обеспечивается высокая скорость работы при низкой точности классификации. При использовании метода BCA [1] для классификации обеспечивается высокая точность при очень больших затратах времени. Применение метода aiNet [1, 2] характеризуется большей точностью классификации, в отличие от CLONALG, но меньшей, чем точность BCA. Кроме того, метод aiNet является более медленным по сравнению с методом CLONALG.

В [3, 4] предложен метод классификации, основным отличием которого является приоритетное последовательное клонирование с конкурентно-целевым отбором клонов, что приводит к сокращению избыточных вычислений, присущих перечисленным иммунным методам. Это приближает его по точности к методу BCA, а по времени, затрачиваемого на классификацию – к методу aiNet. Основными недостатками метода, предложенного в [3, 4], является высокая сложность реализации, не оптимальная мутация, которая приводит к увеличению времени классификации. Предлагаемый в работе метод иммунной классификации объектов является модификацией метода, предложенного в [3, 4]. Изменениям подверглись оператор мутации и способ определения целевых антигенов, что приводит к повышению точности и сокращению времени классификации.

2. Постановка задачи

Дано множество классифицируемых объектов, представленных популяцией антител $AB\{ab_1; \dots; ab_m\}$, и множество классов, представленных популяцией антигенов $AG\{ag_1; \dots; ag_n\}$. Форматом представления антител и антигенов могут быть матрицы признаков, либо множество векторов признаков переменной длины, при этом каждый столбец матрицы (каждый вектор признаков) представляет собой отдельную группу признаков антигена или антитела. Каждая группа признаков характеризуется длиной, т.е. количеством признаков в группе, и весом, т.е. степенью важности данной группы признаков при определении принадлежности к какому-либо классу. При этом веса и длины соответствующих групп признаков одинаковы как для антител, так и для антигенов.

В качестве меры близости между антигенами и антителами используется аффинность [1, 2]:

$$\text{aff}(\text{ag}_i; \text{ab}_j) = (1 + d(\text{ag}_i; \text{ab}_j))^{-1}, \quad (1)$$

где $\text{aff}(\text{ag}_i; \text{ab}_j)$ – аффинность между i -м антигеном и j -м антителом; $d(\text{ag}_i; \text{ab}_j)$ – евклидово расстояние между ними.

Следует отметить, что определение аффинности с помощью (1) применяется только для метрических признаков. Кроме того, выражение (1) используется в простейшем случае, когда классы и объекты характеризуются только одной группой признаков – одним вектором признаков. В случае, когда применяются матрицы признаков, либо совокупность векторов переменной длины, аффинность между антигенами и антителами определяется следующим образом:

$$\text{aff}(\text{ag}_i; \text{ab}_j) = \sum_{c=1}^k w_c (1 + d(\text{ag}_i; \text{ab}_j)_c)^{-1}, \quad (2)$$

здесь k – общее количество групп признаков, т.е. количество столбцов в матрице или векторов переменной длины; w – вес группы признаков; $d(\text{ag}_i; \text{ab}_j)_c$ – евклидово расстояние между признаками данной группы для i -го антигена и j -го антитела.

Цель исследования. Необходимо разработать иммунный метод классификации объектов, устраняющий проблему избыточности вычислений, характерной для методов ВСА [1] и aiNET [1, 2], и проблему больших временных затрат на классификацию, присущую методу ВСА, без потери точности определения классов.

3. Метод иммунной классификации

Предлагаемый метод классификации объектов является модификацией метода, предложенного в [3, 4]. Изменениям подверглась последовательность вызовов иммунных операторов и принципы организации их работы. Предлагаемый метод назван алгоритмом целевого клонального отбора (targeting clonal selection algorithm – TACSA). Формально на уровне иммунных операторов метод TACSA описывается следующим образом:

$$\begin{aligned} \text{TACSA}(\text{AG}, \text{AB}, p) = & \text{AgPresentation}(\text{AG}, \text{AB}) \rightarrow \text{FirstSelection}(\text{AB}) \rightarrow \text{TrgDetection}(\text{AB}) \rightarrow \\ & \rightarrow \text{AB}[\text{Cloning}(\text{ab}, \text{CL}) \rightarrow \text{Mutation}(\text{CL}) \rightarrow \text{TrgPresentation}(\text{ab}_{\text{TRG}}, \text{CL}) \rightarrow \\ & \rightarrow \text{ClonSelection}(\text{CL}) \rightarrow \text{Apoptosys}(\text{ab}, \text{cl})] \text{TermTest}(p), \end{aligned} \quad (3)$$

где $\text{AgPresentation}(\text{AG}, \text{AB})$ – оператор представления антигенов антителам для определения аффинностей; $\text{FirstSelection}(\text{AB})$ – оператор первичного отбора антител; $\text{TrgDetection}(\text{AB})$ – оператор определения целевых антигенов и антител-делегатов; $\text{Cloning}(\text{ab}, \text{CL})$ – клонирование антител, прошедших первичный отбор; $\text{Mutation}(\text{CL})$ – оператор мутации клонов; $\text{TrgPresentation}(\text{ab}_{\text{TRG}}, \text{CL})$ – оператор представления целевых антигенов клонам; $\text{ClonSelection}(\text{CL})$ – конкурентный целевой отбор клонов; $\text{Apoptosys}(\text{ab}, \text{cl})$ – оператор старения антител; $\text{TermTest}(p)$ – процедура проверки критерия останова.

Завершение работы метода TACSA может происходить при выполнении одного из следующих условий: 1) достижение полной специфичности – состояния, при котором все антитела стали специфичными какому-либо антигену в ходе клонирования и мутации; 2) достижение предельного количества популяций антител. Специфичным антигену является антитело, которое в ходе мутации полностью приняло значения признаков антигена. В методе TACSA для повышения скорости обработки популяции антител используются антитела-делегаты. Под антителом-делегатом понимается антитело с хорошей аффинностью к антигенам, которое представляет группу ближайших к нему антител с худшей аффинностью к антигенам. Следует отметить, что операторы представления антигенов, первичного отбора и определения целевых антигенов и антител-делегатов вызываются только один раз для исходной популяции антител и не используются в дальнейшем.

Оператор представления антигенов $\text{AgPresentation}(\text{AG}, \text{AB})$ реализует несколько функций: определение критерия первичного отбора антител и определение аффинностей между

антигенами и антителами (2). Критерием отбора антител является значение средней аффинности между антигенами $CSel$ [3, 4]:

$$CSel = \frac{1}{n} \sum_{i=1}^n aff(AG, ag_i), \quad (4)$$

где $aff(AG, ag_i)$ – аффинность i -го антигена со всеми остальными антигенами, определяемая в соответствии с (2).

Оператор первичного отбора $FirstSelection(AB)$ вызывается для выделения антител исходной популяции, у которых аффинность с популяцией антигенов удовлетворяет следующему условию [3, 4]:

$$aff(AG, ab_i) \geq f \cdot CSel, \quad (5)$$

здесь $aff(AG, ab_i)$ – аффинность i -го антитела с популяцией антигенов AG , f – коэффициент первичного отбора антител, значения которого предлагается выбирать в диапазоне $[0,85; 0,95]$.

Оператор определения целевых антигенов и антител-делегатов $TrgDetection(AB)$ используется для выделения целевых антигенов для антител, прошедших первичный отбор на основании аффинностей, определённых на этапе представления антигенов антителам. Антиген становится целевым для антитела и его клонов в случае выполнения следующего условия:

$$aff(ag_i, ab_j) \geq r \cdot CSel, \quad (6)$$

где $aff(ag_i, ab_j)$ – аффинность i -го антигена с j -м антителом; r – коэффициент сокращения области поиска целевых антигенов, значения которого предлагается выбирать в диапазоне $[1,5; 1,9]$. Если антитело не имеет ни одного целевого антигена – оно не проходит первичный отбор. Для всех антител, не прошедших первичный отбор, выбирается антитело-делегат из числа антител, прошедших первичный отбор. Таким образом, в результате первичного отбора и определения целевых антигенов исходная популяция антител разделяется на две части: множество антител-делегатов, имеющих хотя бы один целевой антиген, над которыми будут производиться дальнейшие действия (клонирование, мутация, старение), и множество антител, не прошедших первичный отбор либо не имеющих ни одного целевого антигена и определивших антитело-делегат по наилучшей аффинности. Антитела, не прошедшие этап первичного отбора, не подвергаются клонированию, мутации и действиям других иммунных операторов. Их принадлежность к какому-либо классу определяется по результатам классификации их антител-делегатов, т.е. антитело, не прошедшее этап первичного отбора, принадлежит тому же классу, что и его антитело-делегат. Антитело-делегаты определяются таким же образом, как и целевые антигены. В случае, если антитело не первичного отбора не может определить антитело-делегат в соответствии с (6), выбор антитела-делегата осуществляется по наилучшей аффинности с каким-либо антителом, прошедшим первичный отбор и имеющим целевые антигены. Использование антител-делегатов сокращает объем избыточных вычислительных операций, возникающих на этапе клонирования и мутации в других иммунных методах.

Оператор клонирования $Cloning(ab, CL)$ используется для клонирования антител, прошедших первичный отбор, при этом применяется пропорциональное клонирование [1], при котором количество клонов, создаваемых антителом, прямо пропорционально его средней аффинности с целевыми антигенами. Следует отметить, что антитела, ставшие специфичными какому-либо антигену, не подвергаются действию операторов клонирования и старения. В других иммунных методах [1, 2] такие антитела называют клетками памяти.

Оператор мутации $Mutation(CL)$ используется для внесения изменений в признаки клонов антител для достижения состояния специфичности клонов какому-либо целевому антигену. При этом важную роль играет коэффициент мутации, который определяет характер изменений признаков клонов. В существующих иммунных методах [1, 2] наилучший

эффект даёт обратно пропорциональная мутация, при которой коэффициент мутации определяется следующим образом:

$$\mu = \text{rand}(0; 1 - \text{aff}(\text{AG}; \text{ab}_i)), \quad (7)$$

где $\text{aff}(\text{AG}; \text{ab}_i)$ – аффинность i -го клонированного антитела с популяцией антигенов.

Данный подход не оптимален, поскольку даже при использовании направленной мутации [1] достижение состояния специфичности клонами во время мутации маловероятно, что обуславливается нулевой нижней границей определения коэффициента мутации. Суть данной проблемы заключается в том, что в случае, если значение коэффициента мутации будет близко к нижней границе – изменения в клонах при мутации будут очень незначительными, в результате чего для достижения состояния специфичности потребуется большое количество популяций антител, что в свою очередь приведёт к увеличению вычислений.

В предлагаемом методе TACSA нижняя граница при определении коэффициента мутации является значением аффинности антитела-предка из предшествующего поколения антител. Определим такую нижнюю границу коэффициента мутации как аффинность дальнего предка. При этом коэффициент мутации предлагается определять в соответствии с выражением:

$$\mu = \text{rand}(1 - \text{aff}(\text{ab}_{\text{TRG}}; \text{ab}'_i); 1 - \text{aff}(\text{ab}_{\text{TRG}}; \text{ab}_i)), \quad (8)$$

где $\text{aff}(\text{ab}_{\text{TRG}}; \text{ab}'_i)$ – аффинность i -го антитела, породившего клонированное антитело текущей популяции с целевыми антигенами, а $\text{aff}(\text{ab}_{\text{TRG}}; \text{ab}_i)$ – аффинность i -го антитела, для клонов которого определяется коэффициент мутации. Следует отметить, что поскольку для клонов, сформированных исходной популяцией антител, дальних предков не существует, при определении коэффициента мутации используется нулевая нижняя граница. Применение аффинностей дальних предков при определении коэффициента мутации начинается со второй популяции антител до завершения работы TACSA. Таким образом, решается проблема предельно малых значений коэффициента мутации, характерная для других иммунных методов [1, 4].

Оператор представления целевых антигенов $\text{TrgPresentation}(\text{ab}_{\text{TRG}}, \text{CL})$ используется для определения аффинностей между мутированными клонами и целевыми антигенами. Таким образом, вместо всей популяции антигенов во взаимодействие с клонами вступают только небольшая группа целевых антигенов, что приводит к снижению количества вычислительных операций.

Оператор $\text{ClonSelection}(\text{CL})$ используется для проведения отбора клонов и определения одного клона с наилучшей аффинностью с одним из своих целевых антигенов. Механизм конкурентно-целевого отбора описан в [3, 4] и не подвергается модификации в TACSA.

Оператор старения $\text{Apoptosis}(\text{ab}, \text{cl})$ используется для замены клонированного антитела клоном, оставшимся в результате конкурентно-целевого отбора. Клонированное антитело заменяется своим клоном в том случае, если его аффинность хуже аффинности клона. В противном случае клон удаляется, и антитело остается для клонирования на следующей популяции.

Процедура проверки критерия останова $\text{TermTest}(p)$ применяется для завершения работы метода, которое может произойти в случае достижения предельного количества популяций антител, либо достижения состояния полной специфичности антител.

Следует отметить, что в TACSA как и в BCA каждое антитело обрабатывается последовательно, т.е. для каждого антитела, прошедшего процедуру первичного отбора, вызывается последовательность операторов клонирования, мутации, представления целевых антигенов клонам, отбора клонов и старения. В методах CLONALG [1] и aiNet [1] происходит обработка всей популяции антител, что повышает сложность их реализации, увеличивает количество вычислительных операций и увеличивает время, затрачиваемое на классификацию.

Процесс классификации с помощью метода TACSA представляется следующим образом:

1. Определение критерия отбора антител (4).
2. Первичный отбор антител (5).
3. Определение целевых антигенов и антител-делегатов (6).
4. Для каждого антитела, прошедшего первичный отбор, выполняется:
 - 4.1. Пропорциональное клонирование.
 - 4.2. Мутация клонов на основе аффинности предыдущих поколений (8).
 - 4.3. Представление целевых антигенов клонам.
 - 4.4. Отбор клонов.
 - 4.5. Старение клонированного антитела.
5. Проверка критерия останова, в случае его достижения – переход к 6, иначе переход к 4.
6. Конец.

В результате работы метода TACSA для множества исходных классифицируемых объектов принадлежности к классам определяются по одному из перечисленных ниже способов:

- специфичные антигенам антитела принадлежат тому же классу, что и антигены;
- не специфичные антигенам антитела принадлежат тому классу, к которому относится большинство его целевых антигенов;
- антитела, не прошедшие процедуру первичного отбора, относятся к тем же классам, что и их антитела-делегаты.

Таким образом, всё множество исходных объектов будет классифицировано независимо от прохождения процедуры первичного отбора.

Экспериментальные результаты

Экспериментальные исследования производилась для методов BCA [1], aiNET [1, 2], CLONALG [1, 2], метода, предложенного в [3, 4], предлагаемого метода TACSA, а также метода kNN [5] с количеством ближайших соседей, равным 3. При исследовании использовались три набора входных данных: первый набор состоял из 100 объектов и 5 классов, в котором каждый объект представлялся вектором из 5 признаков; второй набор – из 300 объектов и 10 классов, в котором объекты представлялись матрицей 3×5 признаков; третий набор содержал 500 объектов, представленных матрицами 5×5 признаков и 15 классами.

В качестве критерия останова использовалось предельно допустимое количество популяций антител (100). Результаты классификации приведены в процентном отношении к ее эталонным результатам. Это значит, что для каждого набора классифицируемых объектов существует заранее известный результат классификации, который является эталонным по точности (100%) для исследуемых методов. При оценке быстродействия среди исследуемых методов был выбран самый медленный, который и является эталонным (100%) по отношению к остальным (таким является метод BCA). После этого для остальных методов классификации проводилось сопоставление их времени работы с временем работы эталонного метода, на основе которого формируется конечное значение. Результаты классификации по точности и временным затратам приведены в таблице.

Таблица

Метод	I набор данных		II набор данных		III набор данных	
	Точность, %	Время, %	Точность, %	Время, %	Точность, %	Время, %
kNN(3)	93	12,78	91	12,52	88	12,39
CLONALG	62	41,24	61	42,01	52	41,67
BCA	95	100,00	94	100,00	91	100,00
aiNET	78	76,39	79	75,98	76	76,03
метод [4,5]	92	65,17	91	64,81	89	65,29
TACSA	96	56,11	96	55,83	95	56,42

По результатам классификации различными методами видно, что иммунные методы классификации требуют для работы больше времени, чем метод kNN, однако некоторые из них превосходят его по точности даже на небольших наборах данных. Среди иммунных

методов наибольшим быстродействием выделяется CLONALG, однако он отличается также и наименьшей точностью при определении классов. ВСА является наиболее медленным методом классификации, однако он даёт очень высокую точность определения принадлежности объектов к классам. Метод, предложенный в [3, 4], по точности классификации сопоставим с методом ВСА, но превосходит его по быстродействию на 35-40 %. Предлагаемый метод TACSA превосходит по быстродействию и по точности большинство иммунных методов, использующихся для классификации. Кроме того, по точности работы метод TACSA превосходит метод kNN в среднем на 4-8%, особенно при работе с большими наборами объектов, и сопоставим только с ВСА.

Выводы

Научная новизна. В предложенном методе классификации объектов TACSA на основе искусственных иммунных систем используются новые подходы к мутации и классификации объектов, наиболее удалённых от классов. Благодаря применению аффинностей дальних предков при определении коэффициента мутации повышается быстродействие метода за счёт сокращения количества популяций антител и объёмов вычислений при определении принадлежности объектов классам. Для объектов, располагающихся на большом расстоянии от классов, используются антитела-делегаты, которые выбираются из множества объектов, прошедших процедуру первичного отбора и имеющих целевые антигены. Принадлежность объектов к классам, не прошедших первичный отбор, определяется принадлежностью их антител-делегатов, что ведет к сокращению объёмов вычислений и ускорению классификации объектов. Кроме того, предложенный метод TACSA более прост в реализации и модификации для решения различных практических задач, что устраняет недостатки метода, изложенного в [3, 4].

Предложенный метод классификации по точности и скорости работы превосходит большинство иммунных методов, являющихся наиболее часто используемыми при решении различных практических задач, в том числе задач классификации и кластеризации данных.

Список литературы: 1. *Dasgupta D.*, Immunological computation, theory and applications / D. Dasgupta, L. F. Nino – CRC Press, 2009. 298 p. 2. *Timmis J.* An Overview of Artificial Immune Systems / J. Timmis, T. Knight, L.N. de Castro, E. Hart // Natural Computation, Springer, 2004. P. 55-86. 3. *Кораблёв Н.М.* Классификация объектов на основе искусственных иммунных систем / Н.М. Кораблёв, А.А. Фомичёв // Системы обработки інформації. 2010. № 6 (87). С. 13-17. 4. *Кораблёв Н.М.* Исследование иммунных операторов в задаче кластеризации объектов / Н.М. Кораблёв, А.А. Фомичёв // Бионика интеллекта. 2010. № 1 (72). С. 70-74. 5. *Fayyad U.M.* Advances in knowledge discovery and data mining, / U.M. Fayyad, G.P. Shapiro / AAAI Press, 1996. 356 p.

Поступила в редколлегию 16.12.2012

Кораблёв Николай Михайлович, д-р техн. наук, профессор кафедры ЭВМ ХНУРЭ. Научные интересы: интеллектуальная обработка информации на основе искусственных иммунных систем. Увлечения и хобби: футбол, плавание, танцы. Адрес: Украина. 61000, Харьков. пр. Ленина, 14, тел. (050)-530-85-72, email: korablev@kture.kharkov.ua

Фомичёв Александр Александрович, ассист. кафедры ЭВМ ХНУРЭ. Научные интересы: моделирование искусственных иммунных систем. Увлечения: музыка, танцы. Адрес: Украина. 61000, Харьков. пр. Ленина, 14, тел. (093)-332-11-73, email: alexandros_1985@mail.ru

РАЗРАБОТКА КОМБИНИРОВАННЫХ МОДЕЛЕЙ ПРОГНОЗИРОВАНИЯ С КЛАСТЕРИЗАЦИЕЙ ВРЕМЕННЫХ РЯДОВ ПО МЕТОДУ БЛИЖАЙШЕГО СОСЕДА

Рассматривается проблема разработки комбинированных моделей прогнозирования временных рядов. Предлагаются комбинированные модели прогнозирования селективного и гибридного типов с предварительной кластеризацией временных рядов по методу ближайшего соседа. Предложенные модели могут быть составными информационных прогнозных систем, а также использоваться для повышения точности и эффективности управления процессом прогнозирования в части учета преимуществ конкурирующих прогнозов.

1. Введение

Принято считать, что на сегодняшний день развитие экономических систем характеризуется высокой степенью изменчивости, которое происходит в условиях неопределенности, неустойчивости и риска. Предпосылкой этого являются тенденции глобализации, усложнение хозяйственных взаимосвязей, рост темпов развития национальных рынков и т.д. В условиях нестабильности экономических рынков применение классических статистических прогнозных моделей и соответствующих методов прогнозирования временных рядов, которые представляют собой экономические показатели, довольно ограничено. Это объясняется также тем, что подавляющее большинство временных рядов не только экономической природы, для которых возникает задача прогнозирования, характеризуется нелинейностью, нестационарностью и неустойчивостью относительно среднего уровня. Многие прогнозные модели, в частности экспоненциальные [1,2], линейные регрессионные [3,4], авторегрессионные типа ARIMA [5] не предназначены для прогнозирования таких временных рядов, так как ограничены требованиями стационарности, нормальности и независимости остатков рядов. Модификации приведенных моделей для прогнозирования нелинейных процессов также имеют недостатки. Нелинейная регрессионная модель характеризуется сложностью идентификации функциональных зависимостей, а применение модели ARIMAX ограничено сложностью расчета многочисленных параметров. Все это обуславливает необходимость разработки новых и модификации известных методов и моделей прогнозирования временных рядов. Для их прогнозирования на финансовом рынке такие модели должны характеризоваться необходимой точностью, гибкостью и прозрачностью в оценке параметров, должны экономить время и преодолевать ограничения, наложенные на классические статистические модели, а также своевременно реагировать на изменения в динамике развития рынка.

В последнее время в области разработки систем прогнозирования сформировались два основных направления исследований:

- создание адаптивных комбинированных и многоуровневых моделей прогнозирования [4-6];
- разработка и использование методов интеллектуального анализа временных рядов [8-10].

Комбинированные модели применяются для повышения точности прогнозирования. Целью управления процессом прогнозирования комбинированных моделей является учет в оценке прогноза особых преимуществ и «полезных» характеристик каждой из прогнозных моделей, которые составляют их основное множество, т.е. отобранных для построения прогноза на определенном шаге. Недостатками комбинированных моделей являются сложность разработки и необходимость компенсации недостатков каждой из моделей основного множества, при этом не теряя их преимуществ.

На сегодняшний день также пользуется популярностью интеллектуальный анализ временных рядов или Time-Series Data Mining, который представляет собой набор инновацион-

ных методов, предназначенных для идентификации скрытых или ассоциативных правил в данных временных рядов, методы нелинейной динамики [11], оптимизации на основе генетических алгоритмов и программирования генетических выражений [12] и т.д. С момента возникновения интеллектуального анализа временных рядов разработано множество специальных методов, которые касаются задач индексации, кластеризации, классификации [8-10], идентификации выбросов и т.п. В последние годы появляется понимание того, что для многих задач интеллектуального анализа данных необходимо применять не быстрый и приближительный, а медленный, но точный и подробный поиск [13], предлагается использовать методы, которые могут быть максимально легко практически применимы.

Автором предложен подход, который базируется на применении в комплексе адаптивных комбинированных моделей прогнозирования и кластеризации временных рядов по методу ближайшего соседа.

Актуальность этих исследований имеет не только практическое значение для прогнозистов, инвесторов и аналитиков, но и теоретическое: разработка концепций и математического инструментария, которые могут составить основу для дальнейших научных исследований.

Целью является рассмотрение основных понятий кластеризации временных рядов, способов представлений кластеров и формул расчета мер близости между ними, а также разработка адаптивных комбинированных моделей прогнозирования, механизмы селекции и гибридизации в которых корректируются на основе предварительной кластеризации временных рядов по методу ближайшего соседа.

2. Постановка задачи прогнозирования временных рядов

Пусть S – дискретное множество. Дискретным временным рядом

$$\{z_i\}_{i=1}^n = \{z_1, z_2, \dots, z_n\} = \{z(t_1), z(t_2), \dots, z(t_n)\}$$

будем называть конечную последовательность измерений, проводимых в дискретные моменты времени $t_i \in S$, $i = \overline{1, n}$ и фиксирующих определенные характеристики изменения состояний исследуемого процесса или явления, t_1 – начальный момент времени. Будем считать, что измерения проводятся через определенные временные интервалы: минута, час, день, неделя, месяц, квартал и т.д. Результаты измерений z_i действительны и могут представлять собой однородные экономические показатели (уровень инфляции, объем выпущенной продукции), финансовые показатели и т.д. В эконометрике часто рассматривают временной ряд как частичную реализацию стохастического процесса, который представляется бесконечной случайной последовательностью. Но в отличие от элементов случайной выборки, значения временного ряда не являются независимыми и одинаково распределенными.

Выделяют следующие основные задачи прогнозирования временных рядов: прогнозирование будущих значений, прогнозирования знаков приростов временных рядов, идентификация моментов локальных экстремумов. В данной работе будем ограничиваться первой задачей. Сформулируем ее так: на основе ретроспективных значений z_n, z_{n-1}, z_{n-m+1} , $m \leq n$ ряда $\{z_i\}_{i=1}^n$, наиболее точно оценить его поведение в будущем в моменты $t_{n+1}, t_{n+2}, \dots, t_{n+\theta}$, т.е. построить последовательность прогнозных значений

$$\{\hat{z}_i\}_{i=n+1}^{n+\theta} = \{\hat{z}_{n+1}, \hat{z}_{n+2}, \dots, \hat{z}_{n+\theta}\},$$

где θ – горизонт прогнозирования, а m – объем ретроспективной выборки. Обозначим через $\hat{z}_\tau(n)$ – прогноз, который рассчитывается в момент t_n (в точке n) на τ точек вперед, $\tau = \overline{1, \theta}$.

Такая функциональная зависимость f , которая позволяет описать поведение временного ряда, называется моделью прогнозирования. Прогноз временного ряда $\{z_i\}_{i=1}^n$ на одну точку вперед можно формально записать в виде $\hat{z}_{n+1} = \hat{z}_1(n) = f(z_{n-m+1}, z_{n-m+2}, \dots, z_n)$. В случае прогнозирования с горизонтом $\theta > 1$ применяется итерационный подход:

$$\hat{z}_{n+1} = \hat{z}_1(n) = f(z_{n-m+1}, z_{n-m+2}, \dots, z_n),$$

$$\begin{aligned} \hat{z}_{n+2} &= \hat{z}_2(n) = f(z_{n-m+2}, z_{n-m+3}, \dots, z_n, \hat{z}_{n+1}), \\ &\vdots \\ \hat{z}_{n+\theta} &= \hat{z}_\theta(n) = f(z_{n-m+\theta}, \dots, \hat{z}_{n+\theta-2}, \hat{z}_{n+\theta-1}), \quad m \leq n. \end{aligned}$$

Необходимо отметить, что разные модели прогнозирования могут иметь различные механизмы расчета прогнозов, к примеру, в них могут учитываться внешние факторы, которые определенным образом влияют на моделируемый процесс. Кроме того, модель может иметь ряд параметров, требующих отдельного оценивания. К таким моделям относятся регрессионные, авторегрессионные, экспоненциальные, адаптивные полиномиальные модели [3,4]. Еще одной важной характеристикой каждой из моделей является объем ретроспективной выборки, используемой для реализации прогнозов.

Построить наиболее точный прогноз означает построить такую модель, которая удовлетворяет соответствующие критерии оценки качества прогнозирования. Для данной задачи такими критериями могут быть среднее абсолютное отклонение, средняя квадратичная погрешность, стандартное отклонение, относительная погрешность и т.д.

3. Основные понятия кластеризации временных рядов

Пусть задан временной ряд $\{z_i\}_{i=1}^n$ как конечная последовательность действительных чисел. Кластером длины m временного ряда $\{z_i\}_{i=1}^n$ будем называть подпоследовательность $\{z_{k_j}\}_{j=1}^m$ данного ряда с m элементов, $n > m$, $k_{j+1} = k_j + 1$ для $j = \overline{1, m-1}$. Иными словами, кластеры любого временного ряда представляют собой его ретроспективные подпоследовательности с m элементов, причем порядок следования элементов в них такой, как и во временном ряду. Для задания кластера необходимо указать индекс начального элемента ряда и длину. Будем обозначать кластеры через $z_{(m)}^N$, где N – индекс начального элемента ряда, включенного в кластер, или порядковый номер, а m – длина. Тогда кластеры временного ряда $\{z_i\}_{i=1}^n$ можно определить таким образом:

$$\begin{aligned} (1, m) \text{ - кластер} &- z_{(m)}^1 = \{z_{k_1^1}, z_{k_2^1}, \dots, z_{k_m^1}\} = \{z_{k_j^1}\}_{j=1}^m, \quad k_{j+1}^1 = k_j^1 + 1, \quad j = \overline{1, m-1}, \quad z_{k_1^1} = z_1, \\ (2, m) \text{ - кластер} &- z_{(m)}^2 = \{z_{k_1^2}, z_{k_2^2}, \dots, z_{k_m^2}\} = \{z_{k_j^2}\}_{j=1}^m, \quad k_{j+1}^2 = k_j^2 + 1, \quad j = \overline{1, m-1}, \quad z_{k_1^2} = z_2, \\ &\vdots \\ (n-m-1, m) &- \text{ кластер} - z_{(m)}^{n-m-1} = \{z_{k_1^{n-m-1}}, z_{k_2^{n-m-1}}, \dots, z_{k_m^{n-m-1}}\} = \{z_{k_j^{n-m-1}}\}_{j=1}^m, \\ &k_{j+1}^{n-m-1} = k_j^{n-m-1} + 1, \quad j = \overline{1, m-1}, \quad z_{k_1^{n-m-1}} = z_{n-m-1}, \\ (n-m, m) \text{ - кластер} &- z_{(m)}^{n-m} = \{z_{k_1^{n-m}}, z_{k_2^{n-m}}, \dots, z_{k_m^{n-m}}\} = \{z_{k_j^{n-m}}\}_{j=1}^m, \quad k_{j+1}^{n-m} = k_j^{n-m} + 1, \\ &j = \overline{1, m-1}, \quad z_{k_1^{n-m}} = z_{n-m}, \text{ т.е. } (s, m) \text{ - кластер для } s = \overline{1, n-m} \text{ в целом определяется так:} \end{aligned}$$

$$z_{(m)}^s = \{z_{k_1^s}, z_{k_2^s}, \dots, z_{k_m^s}\} = \{z_{k_j^s}\}_{j=1}^m, \quad k_{j+1}^s = k_j^s + 1, \quad j = \overline{1, m-1}, \quad z_{k_1^s} = z_s, \quad (1)$$

$(n-m, m)$ – кластер ряда $\{z_i\}_{i=1}^n$ будем называть опорным. Все другие кластеры будем называть неопорными.

Число неопорных кластеров длины m , построенных на основе временного ряда с n элементов, равно $n-m$.

Следует отметить, что существует некоторая неопределенность в терминологии. Термин кластер (pattern) временного ряда, которым в дальнейшем будем пользоваться, применяется в работе [9]. В работе [10] используют термин vector, встречается также термин кусочки (pieces), set и т.д. Для определения опорного кластера применяют также термины последний пригодный вектор (last available vector), история ряда.

Введем следующие представления кластеров:

1. Кластеры могут представляться непосредственно как подпоследовательности элементов входного временного ряда (1).

2. Представление кластеров с помощью знаковой последовательности. На основе ряда $\{z_i\}_{i=1}^n$ построим знаковый ряд $\{\chi_i\}_{i=2}^n$, где $\chi_i = \text{sgn}(z_i - z_{i-1})$, $i = \overline{2, n}$. Тогда знаковые неопорные кластеры будут иметь вид:

$$\chi_{(m)}^2 = \{\chi_{k_1^2}, \chi_{k_2^2}, \dots, \chi_{k_m^2}\} = \{\chi_{k_j^2}\}_{j=1}^m, \quad k_{j+1}^2 = k_j^2 + 1, \quad j = \overline{1, m-1}, \quad \chi_{k_1^2} = \chi_2,$$

$$\chi_{(m)}^3 = \{\chi_{k_1^3}, \chi_{k_2^3}, \dots, \chi_{k_m^3}\} = \{\chi_{k_j^3}\}_{j=1}^m, \quad k_{j+1}^3 = k_j^3 + 1, \quad j = \overline{1, m-1}, \quad \chi_{k_1^3} = \chi_3,$$

$\vdots$

$$\chi_{(m)}^{n-m-1} = \{\chi_{k_1^{n-m-1}}, \chi_{k_2^{n-m-1}}, \dots, \chi_{k_m^{n-m-1}}\} = \{\chi_{k_j^{n-m-1}}\}_{j=1}^m, \quad k_{j+1}^{n-m-1} = k_j^{n-m-1} + 1, \quad j = \overline{1, m-1},$$

$$\chi_{k_1^{n-m-1}} = \chi_{n-m-1}.$$

Знаковый опорный кластер будет иметь вид

$$\chi_{(m)}^{n-m} = \{\chi_{k_1^{n-m}}, \chi_{k_2^{n-m}}, \dots, \chi_{k_m^{n-m}}\} = \{\chi_{k_j^{n-m}}\}_{j=1}^m, \quad k_{j+1}^{n-m} = k_j^{n-m} + 1, \quad j = \overline{1, m-1}, \quad \chi_{k_1^{n-m}} = \chi_{n-m}.$$

Очевидно, что число знаковых неопорных кластеров длины m , построенных на основе временного ряда с n элементов, равно $n - m - 1$.

3. Третий тип представления основан на введении расстояния между элементами кластеров. Пусть построены кластеры первого типа. В каждом кластере вычислим минимальное и максимальное значения и построим нормированные кластеры по правилу:

$$z_{\max}^s = \max\{z_{k_j^s}\}_{j=1}^m, \quad z_{\min}^s = \min\{z_{k_j^s}\}_{j=1}^m, \quad \text{тогда} \quad \delta_{k_j^s} = \frac{z_{k_j^s} - z_{\min}^s}{z_{\max}^s - z_{\min}^s}, \quad s = \overline{1, n-m}, \quad \text{причем} \\ \delta_{k_j^s} \in [0, 1].$$

Таким образом, для каждого $s = \overline{1, n-m}$ получим нормированные неопорные кластеры и нормированный опорный кластер:

$$\delta_{(m)}^s = \{\delta_{k_1^s}, \delta_{k_2^s}, \dots, \delta_{k_m^s}\} = \{\delta_{k_j^s}\}_{j=1}^m, \quad k_{j+1}^s = k_j^s + 1, \quad j = \overline{1, m-1}, \quad \delta_{k_1^s} = \frac{z_{k_1^s} - z_{\min}^s}{z_{\max}^s - z_{\min}^s}.$$

Число нормированных неопорных кластеров длины m , построенных на основе временного ряда с n элементов, равно $n - m$.

Введем понятие меры близости между кластерами, что в случае количественной оценки выражается на основе определенной метрики. Подобие между двумя кластерами определяется на основе метрического расстояния между ними. Каждый кластер может быть представлен точкой в m -мерном пространстве. В зависимости от представления кластеров можно выделить различные метрические расстояния или меры близости. В данной работе ограничимся первым типом представления (1). Пусть заданы множество $z_{(m)}^v$, $v = \overline{1, n-m-1}$ и опорный кластер $z_{(m)}^{n-m}$ ряда $\{z_i\}_{i=1}^n$, тогда меры близости между опорным и всеми неопорными кластерами могут быть определены по известным формулам:

1. $d_0(z_{(m)}^{n-m}, z_{(m)}^v) = \sqrt{\sum_{j=1}^m \left(z_{k_j^{n-m}} - z_{k_j^v} \right)^2}$ – расстояние Евклида.
2. $d_1(z_{(m)}^{n-m}, z_{(m)}^v) = \sum_{j=1}^m \left| z_{k_j^{n-m}} - z_{k_j^v} \right|$ – манхэттенская (городская) метрика.
3. $d_2(z_{(m)}^{n-m}, z_{(m)}^v) = \left(\sum_{j=1}^m \left| z_{k_j^{n-m}} - z_{k_j^v} \right|^p \right)^{\frac{1}{p}}$ – расстояние Минковского.

4. $d_3(z_{(m)}^{n-m}, z_{(m)}^v) = \sqrt{\left(z_{k_j}^{n-m} - z_{k_j}^v \right)^T S_i^{-1} \left(z_{k_j}^{n-m} - z_{k_j}^v \right)}$, $S_i = \text{cov} \left(z_{k_j}^{n-m} - z_{k_j}^v \right)$ – расстояние Махаланобиса.

5. $d_4(z_{(m)}^{n-m}, z_{(m)}^v) = \sum_{j=1}^m I_v^j$, где $I_v^j = \begin{cases} 1, & \left| z_{k_j}^{n-m} - z_{k_j}^v \right| < \varepsilon \\ 0, & \left| z_{k_j}^{n-m} - z_{k_j}^v \right| \geq \varepsilon \end{cases}$ – расстояние Журавлева.

Моделирование временных рядов на основе кластеризации [8, 10] базируется на том, что любой временной ряд может быть представлен как конечная последовательность кластеров одинаковой длины, использующихся в целях прогнозирования и анализа динамики. Для прогнозирования временного ряда сначала на основе некоторой степени близости находят кластер, подобный (closest match) опорному, последний элемент которого представляет собой точку, в которой рассчитывается прогноз. Далее предполагается, что поведение временного ряда, которое наблюдалось после подобного кластера, будет повторяться и после опорного кластера. Недостатком методики является неоднозначность зависимости результатов прогнозирования от выбора меры близости. Кроме того, данный подход для большинства временных рядов, в том числе тех, которые представляют собой экономические процессы, не отмечается высокой точностью.

4. Адаптивные комбинированные модели прогнозирования с кластеризацией временного ряда по методу ближайшего соседа

Постановка задачи. Пусть задано программное множество $\mathfrak{Z}_{PS}$ моделей прогнозирования $f_1, f_2, \dots, f_K$, на основе которых для ряда $\{z_i\}_{i=1}^n$ в точке n могут быть построены оценки будущих элементов временного ряда $\{z_{n+1}^p, z_{n+2}^p, \dots, z_{n+\theta}^p\}$. На основе множества $\mathfrak{Z}_{PS}$ и ретроспективных значений временного ряда $\{z_i\}_{i=1}^n$ построить наиболее точную последовательность прогнозных значений $\hat{Z} = \{\hat{z}_i\}_{i=n+1}^{n+\theta} = \{\hat{z}_{n+1}, \hat{z}_{n+2}, \dots, \hat{z}_{n+\theta}\}$.

Решение задачи. Пусть для временного ряда $\{z_i\}_{i=1}^n$ по правилу (1) построено множество неопорных кластеров одинаковой длины m :

$$z_{(m)}^v = \{z_{k_1}^v, z_{k_2}^v, \dots, z_{k_m}^v\} = \{z_{k_j}^v\}_{j=1}^m, \quad k_{j+1}^v = k_j^v + 1, \quad j = \overline{1, m-1}, \quad v = \overline{1, n-m-1}, \quad z_{k_1}^v = z_v. \quad (2)$$

Обозначим его через $\mathfrak{K}$. Прогноз, который реализуется на основе каждой модели f_p из программного множества $\mathfrak{Z}_{PS}$, $p = \overline{1, K}$ в точках ряда $m, m+1, \dots, n-1$, $m < n$, соответствующих последним элементам неопорных кластеров $z_{(m)}^v$ (2), $v = \overline{1, n-m-1}$ длины m на $\tau = \overline{1, \theta}$ точек вперед, обозначим через:

$$\begin{aligned} \hat{z}_{k_{w+1}}^p &= f_p \left(\{z_{k_j}^v\}_{j=1}^m \right), \quad w = v + m - 1, \quad k_{j+1}^v = k_j^v + 1, \quad j = \overline{1, m-1}, \quad z_{k_1}^v = z_v, \quad v = \overline{1, n-m-1}, \\ \hat{z}_{k_{w+2}}^p &= f_p \left(\{z_{k_j}^v\}_{j=1}^m, \hat{z}_{k_{w+1}}^p \right), \quad w = v + m - 1, \quad k_{j+1}^v = k_j^v + 1, \quad j = \overline{1, m-1}, \quad z_{k_1}^v = z_v, \quad v = \overline{2, n-m-1}, \\ &\vdots \\ \hat{z}_{k_{w+\theta}}^p &= f_p \left(\{z_{k_j}^v\}_{j=1}^m, \hat{z}_{k_{w+1}}^p, \hat{z}_{k_{w+2}}^p, \dots, \hat{z}_{k_{w+\theta-1}}^p \right), \quad w = v + m - 1, \quad k_{j+1}^v = k_j^v + 1, \quad j = \overline{1, m-1}, \\ z_{k_1}^v &= z_v, \quad v = \overline{\theta, n-m-1}, \end{aligned}$$

где $\hat{z}_{k_{w+\tau}}^p$ – прогнозные значения элемента $w + \tau$ v -го кластера, полученное на основе p -й

прогнозной модели. Запись $f_p \left(\{z_{k_j^v}\}_{j=1}^m \right)$ означает, что для построения прогноза по модели

f_p использован соответствующий v -й неопорный кластер. Для упрощения записей будем считать, что $\hat{z}_{k_{w+\tau}}^p = \hat{z}_{w+\tau}^p = \hat{z}_\tau^p(w)$, где $\hat{z}_\tau^p(w)$ – прогноз, рассчитывающийся в точках w на τ точек вперед по p -й модели, которая для построения прогнозного значения использует кластер $z_{(m)}^v$, $v = \tau, n-m-1$ из множества неопорных кластеров $\mathfrak{R}$.

Построим опорный кластер

$$z_{(m)}^{n-m} = \{z_{k_1^{n-m}}, z_{k_2^{n-m}}, \dots, z_{k_m^{n-m}}\} = \{z_{k_j^{n-m}}\}_{j=1}^m, \\ k_{j+1}^{n-m} = k_j^{n-m} + 1, \quad j = \overline{1, m-1}, \quad z_{k_1^{n-m}} = z_{n-m} \quad (3)$$

и на основе некоторой меры близости, например, с помощью расстояния Евклида по методу ближайшего соседа найдем из множества неопорных кластеров $\mathfrak{R}$ подобный опорному.

Неопорный кластер $z_{(m)}^x \in \mathfrak{R}$ называется подобным опорному кластеру $z_{(m)}^{n-m}$, если не существует других неопорных кластеров $z_{(m)}^k$, $k \in [1, n-m-1]$, $k \neq x$, для которых $d(z_{(m)}^{n-m}, z_{(m)}^k) < d(z_{(m)}^{n-m}, z_{(m)}^x)$.

Согласно принципам построения комбинированных моделей есть два принципа расчета прогнозов: селективный и гибридный.

Селективный подход заключается в отборе для каждого значения τ из программного множества $\mathfrak{Z}_{PS}$ единственной модели, которая обеспечивает высокую точность прогнозирования по определенному критерию селекции: В- [4], R-критерий [7]. Параметры критериев селекции, как правило, имеют адаптивный характер. Кроме того, часто для повышения точности прогнозирования критерии отбора применяют не к программному, а к так называемому основному множеству $\mathfrak{Z}_{BS}^\tau$. Это множество состоит из моделей, дающих наиболее точные прогнозы на определенном участке временного ряда, $\mathfrak{Z}_{BS}^\tau \subset \mathfrak{Z}_{PS}$. Отбор моделей в основное множество может осуществляться, например, на основе D-критерия [4].

Прогноз по гибридному подходу рассчитывается как взвешенная сумма прогнозов по всем моделям, составляющим основное множество $\mathfrak{Z}_{BS}^\tau$.

Построим прогноз на основе этих подходов, используя результаты кластеризации временного ряда. Пусть на основе определенной меры близости был определен кластер $z_{(m)}^{x-m+1} \in \mathfrak{R}$, $x \in [m, n-\theta-1]$, подобный опорному кластеру. Последним элементом кластера $z_{(m)}^{x-m+1}$ будет элемент z_x .

Построим основное множество моделей прогнозирования, используя D-критерий. Значение D-критерия в момент t_x можно рассчитать по формуле:

$$D_p(\tau) = \frac{1}{x-c+1} \sum_{j=0}^{x-c} \left(\hat{z}_\tau^p(x-\tau-j) - z_{x-j} \right)^2, \quad (4)$$

где τ – период прогноза; c – период предыстории; $\hat{z}_\tau^p(x-\tau-j)$ – прогноз, который рассчитывается в момент $t_{x-\tau-j}$ (в точке $x-\tau-j$) на τ точек вперед за моделью f_p , $p = \overline{1, K}$. Тогда основное множество моделей для фиксированного τ определяется так:

$$\mathfrak{Z}_{BS}^\tau = \left\{ f_p \in \mathfrak{Z}_{PS} \mid D_p(\tau) \leq \lambda D_{\min}(\tau), p = \overline{1, K} \right\}, \quad D_{\min}(\tau) = \min_{p=\overline{1, K}} D_p(\tau), \quad (5)$$

здесь λ – действительный параметр, который определяется экспериментальным путем.

Обозначим модели, включенные в множество $\mathfrak{Z}_{BS}^\tau$, через $f_1^\tau, f_2^\tau, \dots, f_{L_\tau}^\tau$, L_τ – количество моделей в множестве $\mathfrak{Z}_{BS}^\tau$, $\tau = \overline{1, \theta}$, $L_\tau \leq K$.

Для каждой модели из основного множества $\mathfrak{Z}_{BS}^\tau$ рассчитаем значения В-критерия:

$$B_{x,\tau}^{q_\tau} = (1 - \alpha_B) B_{x-1,\tau}^{q_\tau} e^{q_\tau(x-\tau)}, \quad (6)$$

где $0 < \alpha_B \leq 1$ – параметр сглаживания; а $e^{q_\tau(x-\tau)} = \left| \hat{z}_\tau^{q_\tau}(x-\tau) - z_x \right|$ – абсолютная ошибка прогноза, который рассчитывается в момент $t_{x-\tau}$ на τ точек вперед по моделям $f_{q_\tau}^\tau$, $\tau = \overline{1, \theta}$, $q_\tau = \overline{1, L_\tau}$. Тогда самой точной моделью для фиксированного τ по В-критерию будет считаться такая модель $f^{\tau*}$, для которой обеспечивается минимальное значение критерия, $\min_{q_\tau = \overline{1, L_\tau}} B_{x,\tau}^{q_\tau}$. Прогнозы, рассчитывающиеся на основе моделей $f^{\tau*}$ в точке n на

τ точек вперед, обозначим через $\hat{z}_\tau^*(n)$. Тогда прогноз по комбинированной модели селективного типа по В-критерию селекции (6) с кластеризацией временного ряда по методу ближайшего соседа будет рассчитываться по формуле:

$$\hat{z}_\tau(n) = \alpha \hat{z}_\tau^*(n) + (1 - \alpha) z_{x+\tau}, \quad (7)$$

где $z_{x+\tau}$ – значение временного ряда, которое следует после кластера $z_{(m)}^{x-m+1}$, подобного опорному; $\alpha \in [0, 1]$ – параметр, указывающий на важность учета прогнозных значений отобранной модели в прогнозе.

Применим гибридный подход. Пусть после проведенной кластеризации в точке x для каждого τ были сформированы основные множества $\mathfrak{Z}_{BS}^\tau$ и рассчитаны значения В-критериев $B_{x,\tau}^{q_\tau}$, $q_\tau = \overline{1, L_\tau}$. Обозначим через $\hat{z}_\tau^{q_\tau}(n)$ – прогноз, который рассчитывается в точке n на τ точек вперед по моделям $f_{q_\tau}^\tau$ из основного множества $\mathfrak{Z}_{BS}^\tau$, $q_\tau = \overline{1, L_\tau}$, $\tau = \overline{1, \theta}$. Тогда прогноз по комбинированной модели гибридного типа с кластеризацией по методу ближайшего соседа определяется по формуле:

$$\hat{z}_\tau(n) = \alpha \sum_{q_\tau=1}^{L_\tau} \omega_{q_\tau}^{q_\tau} \hat{z}_\tau^{q_\tau}(n) + (1 - \alpha) z_{x+\tau}, \quad (8)$$

где $\alpha \in [0, 1]$, веса $\omega_{q_\tau}^{q_\tau}$ определяются на основе В-критерия (6) с учетом коэффициента пропорциональности, находящегося из равенства суммы всех весов единице, $\sum_{q_\tau=1}^{L_\tau} \omega_{q_\tau}^{q_\tau} = 1$,

например, как в работе [4]:

$$\begin{aligned} \text{для } L_\tau = 2 \quad \omega_\tau^1 &= \frac{B_{x,\tau}^2}{B_{x,\tau}^1 + B_{x,\tau}^2}, \quad \omega_\tau^2 = \frac{B_{x,\tau}^1}{B_{x,\tau}^1 + B_{x,\tau}^2}, \\ \text{для } L_\tau = 3 \quad \omega_\tau^1 &= \frac{B_{x,\tau}^2 B_{x,\tau}^3}{B_{x,\tau}^1 B_{x,\tau}^2 + B_{x,\tau}^1 B_{x,\tau}^3 + B_{x,\tau}^2 B_{x,\tau}^3}, \\ \omega_\tau^2 &= \frac{B_{x,\tau}^1 B_{x,\tau}^3}{B_{x,\tau}^1 B_{x,\tau}^2 + B_{x,\tau}^1 B_{x,\tau}^3 + B_{x,\tau}^2 B_{x,\tau}^3}, \quad \omega_\tau^3 = \frac{B_{x,\tau}^1 B_{x,\tau}^2}{B_{x,\tau}^1 B_{x,\tau}^2 + B_{x,\tau}^1 B_{x,\tau}^3 + B_{x,\tau}^2 B_{x,\tau}^3} \text{ и т.д.} \end{aligned}$$

После расчета прогноза прогнозная точка $\hat{z}_1(n)$ используется для построения нового опорного кластера $z_{(m)}^{n-m+1} = \{z_{k_1}^{n-m+1}, z_{k_2}^{n-m+1}, \dots, z_{k_{m-1}}^{n-m+1}, \hat{z}_1(n)\}$, $z_{k_1}^{n-m+1} = z_{n-m+1}$, а ста-

рый опорный кластер $z_{(m)}^{n-m}$ становится неопорным, т.е. включается в множество $\mathfrak{R}$, и процесс расчета начитается сначала: находится на основе определенной меры близости подобный опорному кластер, формируются множества $\mathfrak{Z}_{BS}^{\tau}$ для каждого τ , рассчитываются значения В-критерия для каждой прогнозной модели, которая использует подобную неопорную последовательность в качестве ретроспективной информации. Дальше строится прогноз согласно селективному или гибриднему принципу.

6. Выводы и численные результаты

Описанная методика была реализована в программной среде. На основе программного множества из простых экспоненциальных моделей сглаживания (SESM) или адаптивных полиномиальных моделей Брауна 0-го, 1-го и 2-го порядков с параметрами сглаживания 0,2 были построены обычные адаптивные комбинированные модели гибридного типа и селективного типа по В- (6) и R-критериям [7], а также построены модификации данных моделей с предыдущей кластеризацией временного ряда по методу ближайшего соседа. В качестве меры близости было выбрано расстояние Евклида. В каждой точке для каждого τ автоматически формировалось основное множество моделей $\mathfrak{Z}_{BS}^{\tau}$ (5) по D-критерию (4) с параметром $\lambda = 1,9$. Параметр сглаживания в В-критерии (6) $\alpha_B = 0,6$, период предыстории для построения значений В-критерия равен 10. Для тестирования моделей выбраны ежедневные временные ряды цен на сырье: алюминий, бензин, серебро и т.д. за последние 3 года. Всего по 700 измерений. Были рассчитаны средние ошибки прогнозов на $\tau = \overline{1,10}$ точек вперед по описанным методикам (7,8). Точность комбинированных моделей определяется тем, на сколько точнее их прогноз в сравнении с моделью-лидером, т.е. такой моделью из программного множества, для которой в среднем для фиксированного τ получается минимальная ошибка, в данном случае относительная. Так, простая адаптивная комбинированная селективная модель по В-критерию улучшила самую точную модель из программного множества (модель-лидер) только для $\tau = \overline{1,3}$ в среднем на 8,52%, а предложенная модификация данной модели с предыдущей кластеризацией временного ряда ($\alpha = 0,95$, $m = 20$) улучшила модель-лидер только для $\tau = \overline{8,10}$ в среднем на 2,59 %. Подобные результаты получаются и в случае использования R-критерия. Простая адаптивная гибридная модель улучшает модель-лидер для всех $\tau = \overline{1,10}$: для $\tau = \overline{1,5}$ в среднем на 4,71% по сравнению с результатом модели-лидера, для $\tau > 5$ в среднем на 1,62%. Адаптивная гибридная модель с кластеризацией ($\alpha = 0,95$) улучшает модель-лидер для $\tau > 5$ в среднем на 1,97% и простую гибридную модель в среднем на 0,45%.

Научная новизна. В целях повышения точности прогнозирования временных рядов разработана адаптивная комбинированная модель прогнозирования с предыдущей кластеризацией временного ряда по методу ближайшего соседа. Отличие обыкновенной адаптированной комбинированной модели от предложенной отличается тем, что в последней отбор лучших моделей для расчета прогноза по селективному или гибриднему принципу осуществляется не по результатам прогнозирования на участке ряда, предшествующего точке прогноза, так называемом опорном кластере, а по результатам прогнозирования моделей на другом участке (кластере) ряда, который подобный опорному, т.е. метрическое расстояние которого к опорному кластеру минимально.

Практическая ценность работы в том, что предложенные модели могут использоваться в качестве составляющих информационных прогнозных систем, для повышения точности прогнозирования комбинированных моделей с периодом $\tau > 5$ и для повышения эффективности управления процессом прогнозирования в части учета преимуществ конкурирующих прогнозов. Проведенный сравнительный анализ результатов прогнозирования временных рядов цен на сырье с помощью обыкновенных адаптивных комбинированных моделей и комбинированных моделей с предварительной кластеризацией ряда позволяет сделать вывод, что предложенный подход дает возможность добиться более высокой точности в случае прогнозирования с периодом $\tau > 5$.

Список литературы: 1. *Brown Robert G.* Statistical forecasting for inventory control. US: McGraw-Hill Inc., 1959. 223 p. 2. *Holt Charles C.* Forecasting trends and seasonal by exponentially weighted averages // International Journal of Forecasting. 1957. Vol.20, no.1. P.5-10. 3. *Vercellis Carlo* Business intelligence: data mining and optimization for decision making. John Wiley & Sons, Ltd., Publication, 2009. 417 p. 4. *Лукашин Ю.П.* Адаптивные методы краткосрочного прогнозирования временных рядов: Учеб. пособие. М.: Финансы и статистика, 2003. 416 с. 5. *Box G.E.P., Jenkins G.M.* Time series analysis: forecasting and control. San Francisco: Holden-Day, 1976. 575 p. 6. *Берзлев О.Ю., Маляр М.М., Ніколенко В.В.* Адаптивні комбіновані моделі прогнозування біржових показників // Вісник Черкаського держ. технолог. ун-ту. Серія: технічні науки. 2011. № 1. С. 50-54. 7. *Берзлев А.Ю.* Оценка эффективности прогнозирования и принятия решений на финансовом рынке // «Problems of Computer Intellectualization», V.M. Glushkov Institute of Cybernetics of NAS of Ukraine. Kyiv-Sofia: ITHEA, 2012. С. 249-257. 8. *Keogh E., Pazzani M.* An enhanced representation of time series which allows fast and accurate classification, clustering and relevance feedback // 4th Int'l Conference on Knowledge Discovery and Data Mining. 1998 Aug 27-31. New York. P. 239-241. 9. *Singh S.* Pattern Modeling in Time-Series Forecasting // Cybernetics and Systems. An International Journal. 2000. Vol. 31, no. 1. P. 49-65. 10. *Fernández-Rodríguez F., Sosvilla-Rivero S., Andrada-Félix J.* Nearest-Neighbour Predictions in Foreign Exchange Markets // Fundacion de Estudios de Economia Aplicada. 2002. no.5. 36 p. 11. *Peters E. E.* Fractal market analysis: applying chaos theory to investment and economics. John Wiley & Sons, Inc, 1994. 336 p. 12. *Снитюк В.Є.* Прогнозування. Моделі. Методи. Алгоритми: Навчальний посібник. К.: Маклаут, 2008. 364 с. 13. *Chang C.L.E., Garcia-Molina H., Wiederhold G.* Clustering for approximate similarity search in high-dimensional spaces // IEEE Transactions on Knowledge and Data Engineering. 2002. Vol 14, no.4. P. 792-808.

Поступила в редколлегию 12.11.2012

Берзлев Александр Юрьевич, аспирант каф. кибернетики и прикладной математики математического ф-та ГВУЗ «УжНУ». Научные интересы: прогнозирование и интеллектуальный анализ временных рядов. Адрес: Украина, Ужгород, ул. Университетская, 14, к. 313. E-mail: berzlev@gmail.com

УДК 519.7

Ю.С. НОВИКОВ

СТРУКТУРИЗАЦИЯ СОСТАВНЫХ ОБЪЕКТОВ ПРИ ФОРМИРОВАНИИ ПРОЦЕССНОГО ПРЕДСТАВЛЕНИЯ ЗНАНИЙ

Предлагается подход к выполнению структуризации составных объектов как элементов процессного представления знаний. При структуризации выполняется последовательное сравнение простых атрибутов объектов и их совокупности на основе их типов и значений, сравнения уникальных атрибутов на основе специализированных алгоритмов, сравнения вложенных объектов.

1. Актуальность

Процессное представление знаний формируется на основе анализа наборов последовательностей событий, имеющих временные метки, и выявления на основе такого анализа структурированных объектов и последовательностей действий, составляющих процесс, а также причинно-следственных зависимостей между такими действиями. Для решения указанных задач анализа традиционно используются методы data mining и, в последние годы, методы бурно развивающегося направления process mining (интеллектуального анализа процессов). Исходными данными при решении таких задач являются наборы последовательностей событий, фиксирующих поведение некоторых объектов и связанных с ними процессов. Интеллектуальный анализ процессов направлен на выявление последовательности взаимодействия таких объектов с учетом скрытых причинно-следственных связей между ними. Результатом данного анализа является процессное представление знаний, отражающее не просто возможные наборы действий в предметной области, но и зависимости между ними, что создает возможность организации логического вывода на процессных знаниях. В то же время выявленные причинно-следственные связи между действиями процесса отражают лишь «внешнюю» природу процессных знаний. Данные зависимости в значительной степени зависят от природы и структуры объектов, с которыми оперирует процесс. Следовательно, структура таких объектов отражает «внутренние» закономерности

Список литературы: 1. *Brown Robert G.* Statistical forecasting for inventory control. US: McGraw-Hill Inc., 1959. 223 p. 2. *Holt Charles C.* Forecasting trends and seasonal by exponentially weighted averages // International Journal of Forecasting. 1957. Vol.20, no.1. P.5-10. 3. *Vercellis Carlo* Business intelligence: data mining and optimization for decision making. John Wiley & Sons, Ltd., Publication, 2009. 417 p. 4. *Лукашин Ю.П.* Адаптивные методы краткосрочного прогнозирования временных рядов: Учеб. пособие. М.: Финансы и статистика, 2003. 416 с. 5. *Box G.E.P., Jenkins G.M.* Time series analysis: forecasting and control. San Francisco: Holden-Day, 1976. 575 p. 6. *Берзлев О.Ю., Маляр М.М., Ніколенко В.В.* Адаптивні комбіновані моделі прогнозування біржових показників // Вісник Черкаського держ. технолог. ун-ту. Серія: технічні науки. 2011. № 1. С. 50-54. 7. *Берзлев А.Ю.* Оценка эффективности прогнозирования и принятия решений на финансовом рынке // «Problems of Computer Intellectualization», V.M. Glushkov Institute of Cybernetics of NAS of Ukraine. Kyiv-Sofia: ITHEA, 2012. С. 249-257. 8. *Keogh E., Pazzani M.* An enhanced representation of time series which allows fast and accurate classification, clustering and relevance feedback // 4th Int'l Conference on Knowledge Discovery and Data Mining. 1998 Aug 27-31. New York. P. 239-241. 9. *Singh S.* Pattern Modeling in Time-Series Forecasting // Cybernetics and Systems. An International Journal. 2000. Vol. 31, no. 1. P. 49-65. 10. *Fernández-Rodríguez F., Sosvilla-Rivero S., Andrada-Félix J.* Nearest-Neighbour Predictions in Foreign Exchange Markets // Fundacion de Estudios de Economia Aplicada. 2002. no.5. 36 p. 11. *Peters E. E.* Fractal market analysis: applying chaos theory to investment and economics. John Wiley & Sons, Inc, 1994. 336 p. 12. *Снитюк В.Є.* Прогнозування. Моделі. Методи. Алгоритми: Навчальний посібник. К.: Маклаут, 2008. 364 с. 13. *Chang C.L.E., Garcia-Molina H., Wiederhold G.* Clustering for approximate similarity search in high-dimensional spaces // IEEE Transactions on Knowledge and Data Engineering. 2002. Vol 14, no.4. P. 792-808.

Поступила в редколлегию 12.11.2012

Берзлев Александр Юрьевич, аспирант каф. кибернетики и прикладной математики математического ф-та ГВУЗ «УжНУ». Научные интересы: прогнозирование и интеллектуальный анализ временных рядов. Адрес: Украина, Ужгород, ул. Университетская, 14, к. 313. E-mail: berzlev@gmail.com

УДК 519.7

Ю.С. НОВИКОВ

СТРУКТУРИЗАЦИЯ СОСТАВНЫХ ОБЪЕКТОВ ПРИ ФОРМИРОВАНИИ ПРОЦЕССНОГО ПРЕДСТАВЛЕНИЯ ЗНАНИЙ

Предлагается подход к выполнению структуризации составных объектов как элементов процессного представления знаний. При структуризации выполняется последовательное сравнение простых атрибутов объектов и их совокупности на основе их типов и значений, сравнения уникальных атрибутов на основе специализированных алгоритмов, сравнения вложенных объектов.

1. Актуальность

Процессное представление знаний формируется на основе анализа наборов последовательностей событий, имеющих временные метки, и выявления на основе такого анализа структурированных объектов и последовательностей действий, составляющих процесс, а также причинно-следственных зависимостей между такими действиями. Для решения указанных задач анализа традиционно используются методы data mining и, в последние годы, методы бурно развивающегося направления process mining (интеллектуального анализа процессов). Исходными данными при решении таких задач являются наборы последовательностей событий, фиксирующих поведение некоторых объектов и связанных с ними процессов. Интеллектуальный анализ процессов направлен на выявление последовательности взаимодействия таких объектов с учетом скрытых причинно-следственных связей между ними. Результатом данного анализа является процессное представление знаний, отражающее не просто возможные наборы действий в предметной области, но и зависимости между ними, что создает возможность организации логического вывода на процессных знаниях. В то же время выявленные причинно-следственные связи между действиями процесса отражают лишь «внешнюю» природу процессных знаний. Данные зависимости в значительной степени зависят от природы и структуры объектов, с которыми оперирует процесс. Следовательно, структура таких объектов отражает «внутренние» закономерности

ти процессных знаний, которые в значительной степени ограничивают возможные наборы последовательностей действий процесса. В то же время задача выявления структуры таких объектов, а также определения их влияния на зависимости между действиями процесса до настоящего времени не рассматривалась. Поэтому тема данной работы является актуальной.

2. Анализ последних научных публикаций

Вопросы интеллектуального анализа процессов рассматривались в работах [1,2], где основное внимание уделено обнаружению последовательности действий процесса, а также правил, задающих взаимосвязи между ними. В [3] рассмотрены вопросы выявления структуры сложных наборов данных в базах данных.

Таким образом, вопросы выделения зависимостей и правил при процессном представлении знаний, а также структуризации объектов при проведении интеллектуального анализа данных в литературе рассматриваются разрозненно. В то же время иерархически устроенные сложные объекты являются существенным элементом процессного представления знаний, влияющим на последовательность действий процесса.

3. Постановка задачи исследования

Задача данного исследования заключается в разработке подхода к структуризации составных объектов при формировании процессного представления знаний.

Разработка данного подхода требует: выделения иерархии структурных составляющих составных объектов как элементов процессного представления знаний; определения класса методов (подходов), пригодного для сравнения указанных составляющих и выделения их атрибутов.

Результатом структуризации сложных составных объектов является набор шаблонов и правил, отражающих зависимости между его элементами и непосредственно влияющих на структуру процессного представления знаний.

4. Структуризация составных объектов

Сложные составные объекты являются элементами процессного представления знаний, основаны на структурированных данных и содержат иерархию связанных простых объектов, взаимосвязи между которыми влияют на последовательность процессных действий. Следовательно, возможности манипулирования семантически связанными объектами, которые составляют иерархическую структуру, зависят от наборов их атрибутов, возможных шаблонов и правил их взаимодействия. Примерами предметных областей, для которых характерно манипулирование такими объектами, являются медицина, биология, реинжиниринг бизнес-процессов.

Работа со сложными объектами и построение их иерархии широко используется в области объектно-ориентированного программирования. Ключевыми элементами описания сложных структурированных объектов при этом являются: идентификаторы объектов, их атрибуты, классы, а также механизм наследования. При работе с процессными знаниями меняется наполнение указанных элементов.

Обработка таких структурированных объектов основана на выполнении операций с их атрибутами, например сравнение атрибутов для соответствующих элементов объектов. Такое сравнение в значительной степени зависит от типов сравниваемых атрибутов. Так, для атрибутов с простыми типами первоначально выполняется сравнение типов, а затем сравнение значений с использованием некоторой метрики. Аналогично выполняется сравнение группы атрибутов. В то же время при сравнении специфических типов данных этот подход слабо применим. В этом случае используются разработанные пользователем специализированные функции или общие принципы сравнения. Дополнительные проблемы возникают при сравнении атрибутов, отличающихся размерами, смысловым наполнением и т.п. В данном случае необходимы компромиссные решения, которые учитывают указанные составляющие. Сравнение иерархии объектов объединяет все указанные типы сравнений атрибутов (рисунок). Сложность таких сравнений зависит от глубины иерархии объектов. Это требует проведения структуризации таких объектов путем выделения соответствующих шаблонов, иерархий и правил взаимодействия их составляющих на основе выявления различий в структуре объектов.



Сравнение сложных составных объектов и их составляющих

Предлагаемый подход к структуризации сложных иерархических объектов заключается в последовательном определении различий между их элементами согласно схеме, представленной на рисунке. Иерархия функций сравнения согласно предлагаемому подходу выглядит следующим образом. На нижнем уровне выполняется сравнение отдельных атрибутов с простыми типами, что позволяет оценить наличие базовых типов в структуре объекта. Для сравнения совокупности всех простых атрибутов целесообразно использовать обобщенную функцию, которая выделяет отличия для значений простых типов объекта в целом. Следующий уровень сравнения – выявление допустимых отклонений, полученных в результате сравнения простых атрибутов иерархического устроенного объекта. Далее необходимо сравнить объекты, составляющие сложный иерархический объект, на основе сравнения их атрибутов, как это было показано ранее. Завершающий этап предлагаемого подхода заключается в попарном сравнении иерархически устроенных объектов на основе выявления различий в выполненной на предыдущих этапах структуризации.

5. Выводы

Предложен подход к структуризации составных объектов как элементов процессного представления знаний. Он предусматривает выполнение структуризации путем последовательного сравнения простых атрибутов объекта и их совокупности на основе их типов и значений, сравнения уникальных атрибутов на основе специализированных алгоритмов, сравнения вложенных объектов. Предлагаемый подход создает возможность выделения структурных элементов сложных составных объектов методами интеллектуального анализа данных и процессов и тем самым обеспечивает условия для формирования процессного представления знаний.

Список литературы: 1. *Process Mining : Discovery, Conformance and Enhancement of Business Processes* / W. M. P. van der Aalst . N.–Y. : Springer Verlag. 2011. 370p. 2. *Aalst, W.M.P. van der Process mining in web services : the websphere case* / Aalst, W.M.P. van der, Verbeek // *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering*. 2008. №31(3). P. 45-48. 3. *Han J., Nishio S., and Kawano H. Knowledge discovery in object-oriented and active databases*. In F. Fuchi and T. Yokoi, editors, *Knowledge Building and Knowledge Sharing*, pages 221-230. Ohmsha, Ltd. and IOS Press, 1994. 5.

Поступила в редколлегию 16.11.2012

Новиков Юрий Сергеевич, аспирант кафедры ПИС ХНУРЭ. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-446.

МЕТОД АВТОМАТИЗОВАНОЇ РОЗБУДОВИ ТА ОЦІНЮВАННЯ ЯКОСТІ ОНТОЛОГІЙ БАЗ ЗНАНЬ

Розглядається задача автоматичної розбудови базової онтології. Для підвищення ефективності баз знань, ядром яких є онтологія, пропонується зважувати поняття та відношення онтологій. Розробляються методи завдання ваг відповідних елементів та оптимізація структури онтологій баз знань. Пропонується оцінювати якість онтологій на основі ISO 9126.

1. Вступ

Основною компонентою інтелектуальних систем є база знань (БЗ), що формується відповідно до предметної області (ПО), на яку зорієнтоване функціонування цієї системи. Традиційні методи інженерії знань (отримання знань від експерта, інтелектуальний аналіз даних, машинне навчання тощо) не ґрунтуються на системі вивіренних та загальноприйнятих стандартів, тому побудовані на їхній основі БЗ з часом втрачають свою функціональність через низьку ефективність їх функціонування. Як стандарт інженерії знань використовують онтологічний інжиніринг, у результаті застосування якого отримують онтологію бази знань. Онтологія – це детальна формалізація деякої області знань, подана за допомогою концептуальної схеми. Така схема складається з ієрархічної структури понять, зв'язків між ними, теорем та обмежень, які прийняті у певній ПО [1].

Враховуючи наведене вище, під *формальною моделлю онтології* O розуміють:

$$O = \langle C, R, F \rangle, \quad (1)$$

де C – скінченна множина понять (концептів, термінів) ПО, яку задає онтологія O ; $R: C \rightarrow C$ – скінченна множина відношень між поняттями (термінами, концептами) заданої ПО; F – скінченна множина функцій інтерпретації (аксіоматизація, обмеження), заданих на поняттях чи відношеннях онтології O .

Використання онтологій у складі БЗ допомагає вирішити низку проблем методологічного та технологічного характеру, які виникають під час розроблення таких систем. Зокрема для України характерні проблеми полягають у відсутності концептуальної цілісності й узгодженості окремих прийомів та методів інженерії знань; нестачі кваліфікованих фахівців у цій галузі; жорсткості розроблених програмних засобів та їх низькій адаптивній здатності; складності впровадження інтелектуальних систем, що зумовлено психологічними аспектами. Все це свідчить та підтверджує актуальність проблематики досліджень використання онтологій у процесі побудови інтелектуальних систем.

Для того щоб вручну побудувати повну зв'язану онтологію для певної ПО, необхідно затратити достатньо багато часу та ресурсів. Це пояснюється тим, що прикладні онтології повинні містити десятки тисяч елементів, щоб бути придатними для розв'язування широкого кола задач, які виникають у цих ПО. Ручна побудова онтології – це довгий рутинний процес, який до того ж, вимагає ґрунтовних знань ПО та розуміння принципів побудови онтологій. Тому активно розвиваються методи та алгоритми автоматизованої побудови онтологій. Розробимо математичне забезпечення процесу автоматизації побудови онтології, а точніше, її розбудови, оскільки прийнято, що базові терміни та відношення між ними людина-експерт апріорі ввела в онтологію вручну. Таку початкову онтологію називатимемо базовою і позначатимемо $O_{base} = \langle C_b, R_b, F_b \rangle$, тобто побудова онтології починається з моменту, коли в ній вже є якісь дані. Тому такий процес названо розбудовою базової онтології. Формально запишемо:

$$\chi: O_{base} \rightarrow O. \quad (2)$$

Онтологія – це мова науки. Мова науки, як структуроване наукове знання, задає багатопланове ієрархічне утворення, в якому виділяють такі блоки: терміносистема; номенклатура; засоби та правила формування понятійного апарату і термінів. Отже, щоб побудувати

онтологію, необхідно побудувати її терміносистему O_T та номенклатуру O_N . Базова онтологія обов'язково містить частину терміносистеми, тобто $O_{base} \cap O_T \neq \emptyset$. Енциклопедії, термінологічні та тлумачні словники, на підставі яких будують терміносистему ПО, як правило, мають чітку структуру і складаються із словникових статей. Процес побудови номенклатури складніший. Якщо в словниках терміни вже виокремлені, то в наукових текстах (підручники, монографії тощо) їх необхідно виділяти, вести пошук властивостей понять і відношень між поняттями. Отже, потрібна технологія природномовного опрацювання наукового тексту.

Метою дослідження є розроблення методу автоматизованої розбудови базової онтології та оцінювання її якості.

2. Структурна модель концептів та відношень онтології

Нехай задана множина назв відношень $V = \{v_1, v_2, \dots, v_s\}$. Тоді відношення в онтології задається як відображення із C в C з використанням елемента множини V : $R: C \xrightarrow{v} C$, тобто відношення r_i – триплет вигляду:

$$r_i = \langle C_{i_1}, v_{i_j}, C_{i_2} \rangle. \quad (3)$$

Оскільки онтологія формує таксономію понять, то, використовуючи термінологію об'єктно-орієнтованого підходу, кожне поняття являє собою клас. Визначимо поняття як клас з такою структурою:

$$C = \langle N, R^X, R^Y, S, D, A, Ob \rangle, \quad (4)$$

де N – ім'я поняття; R^X – множина відношень, в яких клас C є доменом (областю визначення); R^Y – множина відношень, в яких клас C є множиною значень; S – суперкласи C ; D – підкласи C ; A – аксіоми визначення C ; Ob – екземпляри C .

Отже, розбудовуючи базову онтологію O_{base} , необхідно будувати триплети r_i та нові поняття C , які задаються структурою (4). У цю структуру входить набір аксіом A , однак автоматизовано побудувати такий набір аксіом дуже складно (принаймні автор не знає жодної такої спроби). Тому поки що такий процес виконують вручну.

Для опрацювання природномовних текстів з метою розбудови онтології нами використано готові парсери, а саме Link Grammar Parser [2]. Розроблено шаблони відношень для шести груп: 1) ієрархія, 2) агрегація, 3) функціональні, 4) семіотичні, 5) тотожності, 6) кореляції, подібно як у роботі [3]. На основі цих шаблонів здійснюється пошук відповідних відношень у тексті.

3. Алгоритм розбудови базової онтології

Наша ідея, яка лежить в основі автоматизованої розбудови онтологій, полягає в тому, що опрацьовані тексти із знаннями ПО використовуються для отримання даних для доповнення наявної онтології. У той самий час проміжна онтологія використовується для опрацювання текстів ПО. У результаті отримуємо рекурсивний процес, який можна вважати самонавчанням системи (рис. 1). Навчання може бути як автоматизованим, так і напівавтоматизованим, за допомогою вчителя. По мірі навчання системи необхідність у вчителі відпаде і процес стане повністю автоматизованим. Початкова онтологія із базовими поняттями ПО та загальноживаними термінами повинна бути задана априорі.

Розбудову онтології організуємо у вигляді її навчання на основі наукових текстів заданої ПО, впорядкованих за зростанням складності опрацювання. Міра складності опрацювання тексту може ґрунтуватись на різних критеріях, наприклад, за кількістю невідомих термінів, які зустрічаються у тексті, або за допомогою топологічного порядку дерева наукових праць, які посилаються одна на одну.

Для визначення нових елементів, які можуть бути додані в онтологію, можна використовувати різноманітні методи, переважно засновані на евристичних, що враховують наявні в онтології елементи. Використавши різні методи, отримуємо набір можливих модифікацій онтологій, серед яких потрібно вибрати правильні. Вибір здійснює вчитель або це здійснюється автоматизовано, на основі попереднього навчання.

Евристики, що беруть участь у визначенні нових елементів, можуть мати вигляд продукційних правил або ґрунтуватися на алгоритмах розпізнавання образів, намагаючись доповнити ділянки онтологій пропущеними елементами на основі існуючих шаблонів.

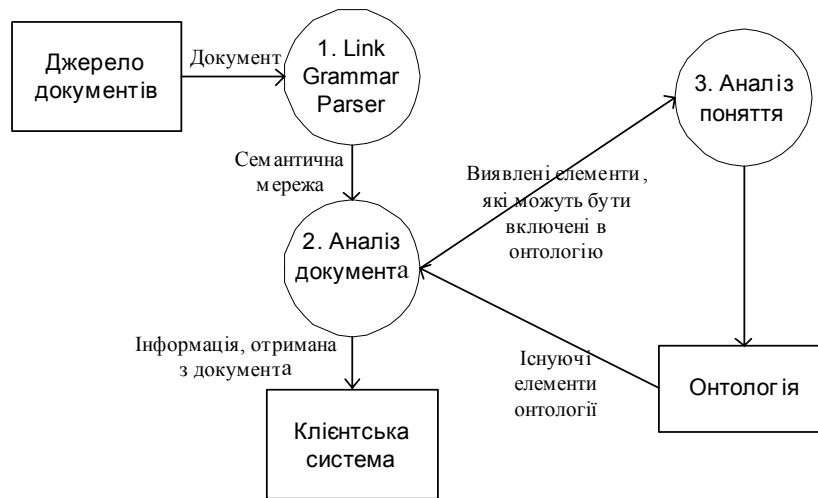


Рис. 1. Діаграма потоків даних автоматизованої розбудови онтології

Розроблений нами алгоритм розвитку базової онтології на основі аналізу природомовних текстів є таким:

- 1) із тексту виділяються семантичні одиниці із прив'язкою до відповідних їм елементів у онтології;
- 2) серед взаємопов'язаних семантичних одиниць виділяються підмножини тих, які утворюють певні семантичні шаблони, котрі можуть утворити нові елементи для онтології;
- 3) семантичні шаблони додаються у масив, по якому після опрацювання текстового документа здійснюється серія проходів. Протягом кожного проходу шаблон розглядається на можливість додавання у онтологію. Якщо такий шаблон є дозволений для додавання політикою побудови онтології, то він поміщається в чергу для розгляду адміністратором або автоматизовано додається залежно від міри довір'я до типу шаблону, встановленою політикою побудови. Проходи здійснюються до того моменту, поки не перестануть додаватися нові елементи або ж фіксовану кількість разів, встановлену політикою побудови;
- 4) черга шаблонів являє собою орієнтований ациклічний граф пропозицій вставки нових елементів у онтологію. Адміністратор розглядає пропозиції з верхнього рівня, якщо пропозицію відхилено, то автоматизовано відхиляються всі пропозиції з нижніх рівнів, які стали можливими завдяки додаванню скасованої в чергу. Якщо пропозицію прийняв адміністратор, то йому надають до розгляду наступні пропозиції поточного рівня; якщо таких не залишилося, то відбувається перехід до наступного рівня. Роль адміністратора може виконуватися евристичним алгоритмом додавання, залежно від політики побудови онтології;
- 5) будь-які дії в онтології логуються у базі даних, підтримується транзакційність і можливість відхилити зміни, починаючи від певного моменту.

Детальніше цей процес нами описано в роботах [4, 5]. Зазначимо, що напрям досліджень автоматизованої розбудови онтологій, БЗ за допомогою природомовних текстів та систем на їхній основі активно розвивається. Зокрема щорічна Європейська конференція зі штучного інтелекту проводить засідання окремої секції з навчання онтологій, на якій розглядає досягнення в галузі їх автоматизованого формування.

Алгоритм працює із семантичною мережею, отриманою після використання Link Grammar Parser. Приклад такої семантичної мережі наведено на рис. 2.

```

INFORMATION SUCH AS THE FOLLOWING IS SHOWN:
linkparser> the quick brown fox jumped over the lazy dog
++++Time 0.04 se
Found 2 linkages (2 had no P.P. violations)
Linkage 1, cost vector = (UNUSED=0 DIS=0 AND=0 LEN=18)

+-----Ds-----+ +-----Js-----+
| +-----A-----+ | +-----Ds-----+
| | +---A---+---Ss---+---Mvp---+ | +---A---+
| | | | | | | | | | |
the quick.a brown.a fox.n jumped.v over the lazy.a dog.n

Press RETURN for the next linkage.

```

Рис. 2. Вигляд текстового документа після використання Link Grammar Parser

Нехай задано речення: „The quick brown fox jumped over the lazy dog”. Дієслово „jumped” відноситься до функціональних відношень $Fun(v,x,y,z)$ (рис. 3). Далі необхідно дослідити, чи існують знайдені поняття в онтології. Можливі п’ять випадків та відповідних дій (таблиця). У випадках 2 та 5 нові поняття вносяться в онтологію, в інших випадках або ігноруються, або заносяться у БД для подальшого аналізу.

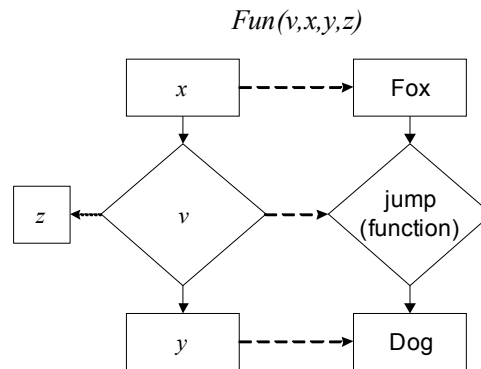


Рис. 3. Відповідність шаблону

Можливі дії під час автоматичної розбудови онтологій

№	Є зв'язок	Є суб'єкт	Є об'єкт	Можлива дія
1	+	+	+	Занести до бази даних відношень
2	+	+	-	Додати невідоме поняття до онтології
3	+	-	-	Аналізувати (Ігнорувати)
4	-	-	-	Ігнорувати
5	+	-	+	Додати невідоме поняття до онтології

Якщо задано речення: „A steel has internal structure”, то в результаті його опрацювання LinkParserом отримаємо відношення “has”, яке належить до зв’язків ієрархічного типу, а його шаблоном є $Hier(a,x,y)$.

4. Зважування елементів онтології

Ефективність адаптації онтології БЗ до особливостей ПО визначають елементи її структури та механізми її адаптації через самонавчання під час експлуатації. Одним з підходів до реалізації таких механізмів є автоматизоване зважування понять БЗ та семантичних зв’язків між ними під час самонавчання. Цю роль виконують ваги важливості понять та зв’язків. Вага важливості поняття (зв’язку) – це числова міра, котра характеризує значущість певного поняття (зв’язку) у конкретній ПО і динамічно змінюється за певними правилами під час експлуатації системи. Запропоновано розширити модель онтології (1), ввівши в її формальний опис ваги важливості понять та відношень [6, 7]. Таку онтологію визначено як:

$$\hat{O} = \langle \hat{C}, \hat{R}, F \rangle, \quad (5)$$

де $\hat{C} = \langle C, W \rangle$, $\hat{R} = \langle R, L \rangle$, своєю чергою W – вага важливості понять C ; L – вага важливості відношень R .

Визначену у такий спосіб онтологію названо адаптивною, тобто такою, що адаптується до ПО за допомогою задання ваг важливості понять та зв'язків між ними.

Для задання ваг важливості відношень для семантичних задач використано дослідження, які провели датські вчені Кнаппе, Бульшков та Андреасен [8]. Вони визначили такі значення ваг відношень: $L_1 = 0,9$; $L_2 = 0,8$; $L_3 = 0,3$; $L_4 = 0,2$. Для відношення тотожності прийнято, що $L_5 = 1$. Відношення кореляції зустрічається лише в ознакових задачах. Для ознакових задач виходимо із евристики, що поняття, які знаходяться нижче за ієрархією, є важливішими, оскільки вони приймають конкретні значення. Виходячи із цього, вважаємо, що $L_1 = L_2 = 1,1$, ваги інших аналогічні як для семантичних задач. Відношення кореляції є двостороннім зв'язком. Його вага дорівнює модулю кореляції між ознаками: $L_6 = |r_{ij}|$.

Методи задання ваг важливості понять є такими:

- експертні оцінки;
- частота вживання понять у наукових текстах;
- за допомогою інтелектуального аналізу даних ПО, в якій функціонує ІСППР.

Метод обчислення ваг понять наведено у роботах [9, 10].

Визначена модель онтології БЗ дає змогу розраховувати ваги своїх елементів у процесі їх додавання, вилучення і використання під час експлуатації системи, завдяки чому реалізує механізм адаптації до заданої користувачем ПО.

Отримані ваги названо вагами базових понять, множину їх позначено W_B . Ці ваги розвинуті на всю онтологію ПО завдяки таксономії понять онтології, відношенню між поняттями та їх інтерпретації. Математично (формально) цей процес записано у вигляді:

$$W_B \xrightarrow{\hat{O}} W. \quad (6)$$

Розвинення ваг на всю онтологію залежить від визначення (аксіоматизації) класів, їх ієрархії (вертикальних зв'язків) та горизонтальних зв'язків. Ми пропонуємо використати дерева рішень для задання початкових ваг понять.

Вершини (ознаки) окремої гілки дерева рішень розміщені на k рівнях. Очевидно, що чим вищий рівень, тим більш значуща ознака, яка міститься на цьому рівні. Крім того, запропоновано ці ваги нормувати, щоб їх сума для кожного класу (гілки дерева рішень) дорівнювала 1.

Ваги визначаються як відношення різниці $(k+1)$ рівня дерева та рівня, на якому міститься ознака, до суми всіх рівнів гілки, тобто: $w_i = \frac{k+1-i}{\sum_{j=1}^k j} = \frac{2(k+1-i)}{(1+k)k}$.

Пропонуємо такий метод визначення ваг понять на всю онтологію. Спочатку ваги всіх ознак дорівнюють 0. Для ознак, які беруть участь у дереві рішень, для відповідного класу до початкової ваги додаємо вагу, отриману на основі дерева. Всі інші обчислюємо для онтології відповідної задачі згідно з формулою:

$$W_j = \sum_{R^x} L_{ij} \cdot W_i + \sum_{R^y} W_k / L_{jk}. \quad (7)$$

У загальному випадку (7) є системою лінійних алгебраїчних рівнянь. Однак в окремих випадках (7) є послідовністю лінійних співвідношень.

5. Оптимізація онтологій та критерії якості її оцінювання

Автоматизована розбудова онтології призводить до виникнення деяких недоліків у її структурі та змісті, невідповідності її наповнення інформаційним потребам користувача. Тому такі системи необхідно «укомплектовувати» набором процедур оптимізації онтології.

Критерії оптимізації сформовано відповідно до стандарту якості ISO 9126 [11]. Згідно з цим стандартом характеристиками якості є:

1. *Функціональна придатність* залежить від повноти та належної побудови онтології, наскільки точно вона описує специфіку ПО та задач, які у ній виникають. Своєю чергою,

повнота онтологій залежить від вміння давати правильні відповіді на запити до неї, а це залежить від того, чи вміє система оцінювати новизну знань, що пропонується додавати до онтології. Мірою якості функціональної придатності буде відсоток нетривіальних (ненульових) правильних відповідей на запити до онтології, тобто $\chi_1 = \frac{N_q^p}{N_q} \cdot 100\%$. Визначення функціональної придатності є однією з базових характеристик онтологій.

2. *Коректність (або достовірність)* функціонування БЗ – це відсоток достовірно розв’язаних задач. Це основна характеристика якості БЗ. Отже, $\chi_2 = \frac{N_z^p}{N_z} \cdot 100\%$.

3. *Використовуваність ресурсів (або ресурсна економічність)* у стандартах відображається зайнятістю ресурсів центрального процесора, оперативної, зовнішньої та віртуальної пам’яті, каналів введення-виведення, терміналів і каналів зв’язку. Для покращення цієї характеристики розглядають оптимізаційну задачу, критерієм якої є мінімізація фізичного об’єму пам’яті, що займає онтологія. З іншого боку, очевидно, що онтологія займає найменший об’єм пам’яті, якщо в ній немає жодного поняття. Тому введено певну порогову величину на займаний онтологією об’єм пам’яті.

4. *Практичність* – важкоформалізоване поняття, яке визначає функціональну придатність і корисність застосування для певних користувачів. У цю групу показників входять субхарактеристики, які з різних аспектів відображають функціональну зрозумілість, зручність освоєння, системну ефективність і простоту використання онтологій. Така придатність ґрунтується на цілісності онтології, тобто відсутності в її тілі взаємозаперечувальних тверджень та дублювання, а також на збалансованості ПО, яка полягає у рівномірному поданні її окремих підрозділів у онтології.

5. *Супроводжуваність* відображається зручністю та ефективністю виправлення, удосконалення або адаптації структури та змісту онтології БЗ залежно від змін у зовнішньому середовищі застосування, а також у вимогах і функціональних специфікаціях замовника.

6. *Мобільність* характеризується тривалістю і трудомісткістю інсталяції БЗ, адаптації та заміщення у разі перенесення на інші апаратні та операційні платформи. Критерієм мобільності є швидкодія, яка виражається часом відгуку на зовнішнє звертання (час реакції на зміну параметрів зовнішнього середовища, до яких чутлива система).

Зважаючи на наведені вище критерії, метод оптимізації онтології передбачає задачу оптимізації її структури та змісту: 1) усунення паралельних ребер, дублювання вершин з однаковими параметрами та інших особливостей структури графу онтології, які можуть порушити її цілісність і знижують ефективність функціонування інтелектуальних систем (задача оптимізації структури онтології); 2) оптимізація змістової частини онтології з метою збільшення її швидкодії та інформаційної насиченості за заданих обмежень на фізичний об’єм пам’яті системи. Розв’язування цих задач рознесено у часі, причому, щоб зберегти цілісність онтології, спершу виконують її структурну перевірку і лише потім – оптимізацію змістової частини послідовною редукцією її графу до виконання вимог вибраних критеріїв через максимізацію суми ваг вершин та ребер такого графу.

В основу задачі мінімізації структури графу онтології покладено типову оптимізаційну задачу теорії графів про відшукування мінімального кістяка, яка полягає у пошуку кістяка мінімальної ваги у зваженому графі. Задача забезпечення узгодженості в структурі графу ефективно розв’язується методом резолюцій. Задача оптимізації змісту зводиться до оберненої задачі про наплічник. Нехай онтологія складається з n елементів загальним об’ємом пам’яті M . У ролі «наплічника» виступає певна задана частка об’єму, наприклад $N = 0,1 \cdot M$, до якої слід віднести найменш цінні елементи (поняття з мінімальною вагою важливості та максимальним об’ємом) для подальшого їх вилучення. Тоді необхідно

максимізувати: $\sum_{i=1}^n \frac{1}{W_i} x_i \rightarrow \max$, таких елементів, для яких $\sum_{i=1}^n m_i x_i \leq N$, де

$$\begin{cases} x_i = 0, & \text{якщо поняття } C_i \text{ залишається,} \\ x_i = 1, & \text{якщо поняття } C_i \text{ вилючається,} \end{cases} \quad m_i - \text{об'єм пам'яті, який займає елемент } C_i.$$

Для розв'язування цієї задачі використано жадібний алгоритм. Детальніше задачі оптимізації онтологій наведено у [9].

Висновки

Розглянуто метод автоматичної розбудови базової онтології з використанням програмного забезпечення Link Grammar Parser. Здійснено класифікацію типів відношень. Розроблено математичне забезпечення та алгоритм визначення типу відношень, які зустрічаються у наукових текстах. Для адаптації онтології бази знань до задач, які нею розв'язуються, запропоновано зважувати елементи онтологій. Розроблено метод задання відповідних ваг елементів, що, у свою чергу, дає змогу оптимізовувати зміст та структуру онтологій. Запропоновано використати ISO 9126 для оцінювання якості онтологій. Подальшим дослідженням стане задача оцінювання новизни знань, які пропонуються додавати в онтології.

Список літератури: 1. Gruber T. A translation approach to portable ontologies / T. Gruber // Knowledge Acquisition. 1993. № 5 (2). Р. 199–220. 2. Link Grammar Homepage [Електронний ресурс]. Режим доступу: <http://bobo.link.cs.cmu.edu/link/>. 3. Найханова Л.В. Технология создания методов автоматического построения онтологий с применением генетического и автоматного программирования / Л.В. Найханова. Улан-Удэ: Изд-во БНЦ СО РАН, 2008. 244 с. 4. Литвин В.В. Метод автоматичної розбудови адаптивної онтології / В.В.Литвин, Д.І.Угрин // Вісник Національного технічного університету «ХПІ». Харків. 2011. № 10. С. 75–82. 5. Литвин В.В. Автоматизація процесу розвитку базової онтології на основі аналізу текстових ресурсів / В.В.Литвин // Вісн. Нац. ун-ту «Львівська політехніка». Серія: Інформаційні системи та мережі. 2010. № 673. С. 319–325. 6. Литвин В.В. Мультиагентні системи підтримки прийняття рішень, що базуються на прецедентах та використовують адаптивні онтології / В.В.Литвин // Радіoeлектроніка, інформатика, управління. 2009. №2(21). С. 120–126. 7. Литвин В.В. Інтелектуальні агенти пошуку релевантних прецедентів на основі адаптивних онтологій / В.В.Литвин // Математичні машини і системи. Київ. 2011. №3. С. 66–72. 8. Knappe R. Perspectives on Ontology-based Querying [Електронний ресурс] / R.Knappe, H. Bulskov, T. Andreassen // International Journal of Intelligent Systems, 2004. Режим доступу: <http://akira.ruc.dk/~knappe/publications/ijis2004.pdf>. 9. Литвин В.В. Бази знань інтелектуальних систем підтримки прийняття рішень / В.В.Литвин; Міністерство освіти і науки, молоді та спорту України, Національний університет «Львівська політехніка». Львів: Видавництво Львівської політехніки, 2011. 240 с. 10. Даревич Р.Р. Оцінка подібності текстових документів на основі визначення інформаційної ваги елементів бази знань / Р.Р.Даревич, Д.Г.Досин, В.В.Литвин, З.Т.Назарчук // Штучний інтелект. Донецьк. 2006. № 3. С. 500–509. 11. ISO/IEC 9126:1991. Information technology – Software product evaluation – Quality characteristics and guidelines for their use. 1991. 39 p.

Надійшла до редколегії 14.11.2012

Литвин Василь Володимирович, д-р техн. наук, доцент кафедри інформаційних систем Національного університету „Львівська політехніка”. Наукові інтереси: побудова інтелектуальних систем підтримки прийняття рішень. Львів, вул. Бандери, 12, тел. (032) 258-25-38.

Гопяк Марія Ярославівна, аспірантка кафедри інформаційних систем Національного університету „Львівська політехніка”. Наукові інтереси: оцінювання якості баз знань інтелектуальних систем. Львів, вул. Бандери, 12, тел. (032) 258-25-38.

Демчук Андрій Богданович, аспірант кафедри інформаційних систем Національного університету „Львівська політехніка”. Наукові інтереси: побудова інтелектуальних систем для людей з вадами зору. Львів, вул. Бандери, 12, тел. (032) 258-25-38.

ИССЛЕДОВАНИЕ ЗАДАЧ ПРИНЯТИЯ РЕШЕНИЙ В СИСТЕМАХ УПРАВЛЕНИЯ ОГРАНИЧЕННЫМИ РЕСУРСАМИ

Рассматриваются методы решения задач, направленных на оптимизацию в системе управления с ограниченными ресурсами. Выделяются преимущества и недостатки традиционных и эволюционных методов.

1. Введение

В последнее десятилетие во всем мире используются подходы экономико-математического моделирования для оценки эффективности управления цепями поставок и производственного менеджмента. В большинстве случаев для этого применяются модели линейного и целочисленного линейного программирования, где в качестве критериев принимают либо прибыль, полученную предприятием при реализации той или иной производственной программы, либо издержки, связанные с производством конечной продукции. Основным недостатком этого подхода является жесткое закрепление производственных ресурсов при выполнении всего комплекса работ на всем временном интервале планирования. В данной работе рассмотрены методы, позволяющие преодолеть это ограничение, разрешив перераспределять производственные мощности в процессе реализации производственной программы. Предложены получаемые при этом модели оптимального управления ресурсами предприятия и методы их анализа.

Актуальность исследования. Создать такую организацию, которая обеспечит выпуск высококачественной продукции (или услуги) в нужном объеме и ассортименте, без проведения исследований невозможно. Особую роль здесь играют исследования систем управления. Эта проблема была актуальной всегда, однако до недавнего времени она в большей степени решалась в рамках математических дисциплин, таких как теория вероятностей, математическая статистика, логика, теория множеств и др.

Целью данного исследования является изучение эффективных методов принятия решений, их сравнение, определение лучшего оптимизационного метода для решения задач управления.

2. Традиционные методы решения

В начале, для сравнения, приведем описание линейных алгоритмов. Одним из них, вероятно, самым популярным, является симплекс-метод.

Симплекс-метод – алгоритм решения оптимизационной задачи линейного программирования путём перебора вершин выпуклого многогранника в многомерном пространстве. Данный метод, имеющий несколько различных форм (модификаций), был разработан в 1947 году Г. Данцигом.

Задача линейного программирования состоит в том, что необходимо максимизировать или минимизировать некоторый линейный функционал на многомерном пространстве при заданных линейных ограничениях.

Следует указать, что каждое из линейных неравенств на переменные ограничивает полупространство в соответствующем линейном пространстве. В результате все неравенства ограничивают некоторый многогранник (возможно, бесконечный), называемый также полиэдральным конусом. Уравнение $W(x) = c$, где $W(x)$ — максимизируемый (или минимизируемый) линейный функционал порождает гиперплоскость $L(c)$. Зависимость от c порождает семейство параллельных гиперплоскостей. Тогда экстремальная задача приобретает следующую формулировку требуется найти такое наибольшее c , чтобы гиперплоскость $L(c)$ пересекала многогранник хотя бы в одной точке. Заметим, что пересечение оптимальной гиперплоскости и многогранника будет содержать хотя бы одну вершину, причём их будет более одной, если пересечение содержит ребро или k -мерную грань. Поэтому максимум функционала можно искать в вершинах многогранника.

Принцип симплекс-метода состоит в том, что выбирается одна из вершин многогранника, после чего начинается движение по его рёбрам от вершины к вершине в сторону увеличения значения функционала. Когда переход по ребру из текущей вершины в другую вершину с более высоким значением функционала невозможен, считается, что оптимальное значение с найдено.

Для решения системы все неизвестные произвольно подразделяют на базисные и свободные. Число базисных переменных определяется числом линейно-независимых уравнений. Остальные неизвестные свободные. Им придают произвольные значения и подставляют в систему. Любому набору свободных неизвестных можно придать бесчисленное множество произвольных значений, которые дадут бесчисленное множество решений. Если все свободные неизвестные приравнять к нулю, то решение будет состоять из значений базисных неизвестных. Такое решение называется базисным.

В теории линейного программирования существует теорема, которая утверждает, что среди базисных решений системы можно найти оптимальное, а в некоторых случаях и несколько оптимальных решений, но все они обеспечат экстремум целевой функции. Таким образом, если найти какой-либо базисный план, а затем улучшить его, то получится оптимальное решение. На этом принципе и построен симплекс-метод.

Последовательность вычислений симплекс-методом можно разделить на две основные фазы:

- нахождение исходной вершины множества допустимых решений (нахождение базисного решения),
- последовательный переход от одной вершины к другой, ведущий к оптимизации значения целевой функции (последовательное улучшение найденного на первом этапе базисного решения).

При этом в некоторых случаях исходное решение очевидно или его определение не требует сложных вычислений, например, когда все ограничения представлены неравенствами вида «меньше или равно» (тогда нулевой вектор совершенно точно является допустимым решением, хотя и, скорее всего, далеко не самым оптимальным). В таких задачах первую фазу симплекс-метода можно вообще не проводить. Симплекс-метод, соответственно, делится на однофазный и двухфазный.

3. Генетические алгоритмы

Предложенные сравнительно недавно — в 1975 году — Джоном Холландом генетические алгоритмы (ГА) основаны на принципах естественного отбора Ч. Дарвина. ГА относятся к стохастическим методам. Эти алгоритмы успешно применяются в различных областях деятельности (экономика, физика, технические науки и т.п.). Генетические алгоритмы относят к области мягких вычислений. Это понятие объединяет такие области, как нечеткая логика, нейронные сети, вероятностные рассуждения, сети доверия и эволюционные алгоритмы, которые дополняют друг друга и используются в различных комбинациях или самостоятельно для создания гибридных интеллектуальных систем. Генетические алгоритмы — это адаптивные методы поиска, которые в последнее время используются для решения задач оптимизации. В них применяются как аналог механизма генетического наследования, так и аналог естественного отбора. При этом сохраняется биологическая терминология в упрощенном виде и основные понятия линейной алгебры.

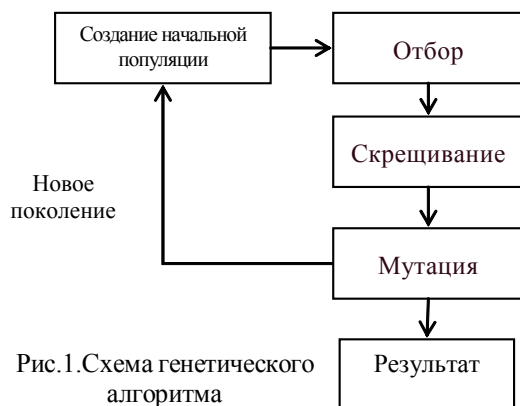


Рис.1.Схема генетического алгоритма

Основной идеей генетических алгоритмов (рис.1) является организация «борьбы за существование» и «естественного отбора» среди пробных решений, которые выбираются случайным образом на требуемом отрезке. Поскольку генетические алгоритмы используют биологические аналогии, то и применяющаяся терминология напоминает биологическую. Так, одно пробное решение называется особью или хромосомой, а набор всех пробных решений — популяцией. Каждая особь

оценивается с использованием «функции приспособленности», в результате чего с каждым генотипом ассоциируется определенное значение («приспособленность»), которое определяет, насколько хорошо фенотип, который им описывается, решает поставленную задачу.

Далее наступает процесс «размножения». Этот процесс в разных алгоритмах реализован по-разному, все зависит от представления данных. Главное требование к размножению — чтобы потомок или потомки имели возможность унаследовать черты обоих родителей, «смешав» их каким-либо способом.

Также возможен этап «мутации». Он подразумевает изменение особи по определенной формуле с целью улучшить ее приспособленность аналогично случайным мутациям в природе. В случае ухудшения результата особь остается неизменной. Это позволяет привнести в результат сторонние значения, которые вероятно хорошо скажутся на следующей популяции.

На этапе отбора должны быть выделены из популяции наиболее приспособленные особи. Обычно их количество задается параметром в начале расчетов. Все особи, не прошедшие отбор, считаются умершими и далее не учитываются. А оставшиеся становятся родителями для следующей итерации алгоритма.

Генетический алгоритм – это простая модель эволюции в природе, реализованная в виде компьютерной программы. Результатом его работы является не оптимальный результат, а просто лучший для данного количества итераций, но данный результат часто является удовлетворительным для сложных систем.

К преимуществам генетических алгоритмов можно отнести простоту их реализации, параллельный поиск решения, что позволяет решить проблему локальных оптимумов, и довольно высокую скорость работы. Но возможен исход, когда все представители популяции идентичны, что останавливает дальнейшую эволюцию и начинается вырождение популяции. Также требуется определить систему кодирования особей и способы их смешивания, что бывает проблематично в зависимости от поставленной задачи.

4. Муравьиные алгоритмы

Имитация самоорганизации муравьиной колонии составляет основу муравьиных алгоритмов оптимизации (рис.2). Колония муравьев может рассматриваться как многоагентная система, в которой каждый агент (муравей) функционирует автономно по очень простым правилам. В противовес почти примитивному поведению агентов, поведение всей системы получается на удивление разумным.

Муравьиные алгоритмы серьезно исследуются европейскими учеными с середины 90-х годов. На сегодняшний день уже получены хорошие результаты для оптимизации таких сложных комбинаторных задач, как задача коммивояжера, задача оптимизации маршрутов грузовиков, задача раскраски графа, квадратичная задача о назначениях, задача оптимизации сетевых графиков, задача календарного планирования и многие другие. Особенно эффективны муравьиные алгоритмы при динамической оптимизации процессов в распределенных нестационарных системах, например, трафиков в телекоммуникационных сетях.

Идея муравьиного алгоритма – моделирование поведения муравьев, связанного с их способностью быстро находить кратчайший путь от муравейника к источнику пищи и адаптироваться к изменяющимся условиям, находя новый кратчайший путь. При своём движении муравей метит путь феромоном, и эта информация используется другими муравьями для выбора пути. Это элементарное правило поведения и определяет способность муравьев находить новый путь, если старый оказывается недоступным. Например, на оптимальном доселе пути возникает преграда. В этом случае необходимо определение нового оптимального пути. Дойдя до преграды, муравьи с равной вероятностью будут обходить её справа и слева. То же самое будет происходить и на обратной стороне преграды. Однако те муравьи, которые случайно выберут кратчайший путь, будут быстрее его проходить, и за несколько передвижений он будет более обогащён феромоном. Поскольку движение муравьев определяется концентрацией феромона, то следующие будут предпочитать именно этот путь, продолжая обогащать его феромоном до тех пор, пока этот путь по какой-либо причине станет недоступен.

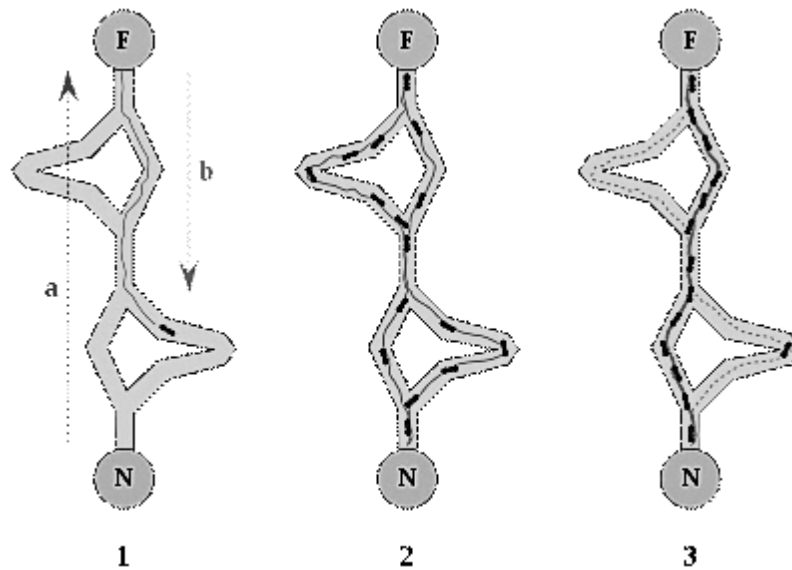


Рис. 2. Схема муравьиного алгоритма

Как было сказано выше, муравьиный алгоритм моделирует многоагентную систему. Ее агентов в дальнейшем будем называть муравьями. Как и настоящие муравьи, они довольно просто устроены: для выполнения своих обязанностей они требуют небольшое количество памяти, а на каждом шаге работы выполняют несложные вычисления.

Каждый муравей хранит в памяти список пройденных им узлов. Этот список называют списком запретов (tabu list) или просто памятью муравья. Выбирая узел для следующего шага, муравей «помнит» об уже пройденных узлах и не рассматривает их в качестве возможных для перехода. На каждом шаге список запретов пополняется новым узлом, а перед новой итерацией алгоритма – т.е. перед тем, как муравей вновь проходит путь – он опустошается.

Кроме списка запретов, при выборе узла для перехода муравей руководствуется «привлекательностью» ребер, которые он может пройти. Она зависит, во-первых, от расстояния между узлами (т.е. от веса ребра), а во-вторых – от следов феромонов, оставленных на ребре прошедшими по нему ранее муравьями. Естественно, что в отличие от весов ребер, которые являются константными, следы феромонов обновляются на каждой итерации алгоритма: как и в природе, со временем следы испаряются, а проходящие муравьи, напротив, усиливают их.

Пусть муравей находится в узле i , а узел j – это один из узлов, доступных для перехода: $j \in S_i$. Обозначим вес ребра, соединяющего узлы i и j , как w_{ij} , а интенсивность феромона на нем – как t_{ij} . Тогда вероятность перехода муравья из i в j будет равна:

$$p_{ij} = \frac{t_{ij}^{\alpha} + \frac{1}{w_{ij}^{\beta}}}{\sum_{l \in S_i} \left(t_{il}^{\alpha} + \frac{1}{w_{il}^{\beta}} \right)} \quad (1)$$

где α и β – это регулируемые параметры, определяющие важность составляющих (веса ребра и уровня феромонов) при выборе пути. Очевидно, что при $\alpha=0$ алгоритм превращается в классический жадный алгоритм, а при $\beta=0$ он быстро сойдется к некоторому субоптимальному решению. Выбор правильного соотношения параметров является предметом исследований, и в общем случае производится на основании опыта.

После того, как муравей успешно проходит маршрут, он оставляет на всех пройденных ребрах след, обратно пропорциональный длине пройденного пути:

$$\Delta_{ij} = \begin{cases} \frac{k}{L}, (ij) \in P \\ 0, (ij) \notin P \end{cases} \quad (2)$$

где L – длина пути, а k – регулируемый параметр. Кроме этого, следы феромона испаряются, т.е. интенсивность феромона на всех ребрах уменьшается на каждой итерации алгоритма. Таким образом, в конце каждой итерации необходимо обновить значения интенсивностей:

$$t_{ij} = (1 - \epsilon) \cdot t_{ij} + \Delta t_{ij} \quad (3)$$

Результаты первых экспериментов с применением муравьиного алгоритма для решения задачи коммивояжера были многообещающими, но далеко не лучшими по сравнению с уже существовавшими методами. Однако простота классического муравьиного алгоритма (названного «муравьиной системой») оставляла возможности для доработок – и именно алгоритмические усовершенствования стали предметом дальнейших исследований Марко Дориги и других специалистов в области комбинаторной оптимизации. В основном, эти усовершенствования связаны с большим использованием истории поиска и более тщательным исследованием областей вокруг уже найденных удачных решений.

Эффективность муравьиных алгоритмов сравнима с эффективностью общих метаэвристических методов, а в ряде случаев – и с проблемно-ориентированными методами. Наилучшие результаты муравьиные алгоритмы показывают для задач с большими размерностями областей поиска. Муравьиные алгоритмы хорошо подходят для применения вместе с процедурами локального поиска, позволяя быстро находить начальные точки для них.

Наиболее перспективными направлениями дальнейших исследований следует считать анализ способа выбора настраиваемых параметров алгоритмов. В последние годы предлагаются различные способы адаптации параметров алгоритмов «на лету». Поскольку от выбора параметров сильно зависит поведение муравьиных алгоритмов, именно к этой проблеме обращено наибольшее внимание исследователей на данный момент.

Выводы

Изучены эффективные методы принятия решений. Важно понять, что выбор того или иного метода зависит от типа решаемой задачи.

В противовес традиционному симплекс-методу генетический и муравьиный алгоритмы представляют собой сложную самоорганизующуюся систему и перспективны в применении с использованием современных информационных технологий.

Научная новизна: результатом проведенного исследования является решение многокритериальной задачи с применением эволюционных алгоритмов.

Практическая значимость: предложенная информационная подсистема позволяет управлять предприятием с ограниченными ресурсами с применением современных информационных технологий.

Список литературы: 1. Гвоздинский А.Н., Якимова Н.А., Губин В.А. Методы оптимизации в системах принятия решений. Харьков, ХНУРЭ. 2006. 325 с. 2. Бондаренко М.Ф., Гвоздинский А.Н. Оптимизационные задачи в системах принятия решений. Харьков, ХТУРЭ. 1998. 3. Гвоздинский А.Н., Клименко Е.Г. Применение генетических алгоритмов для решения оптимизационных задач. Харьков, ХНУРЭ. 2001. С.390-391. 4. Гвоздинский А.Н., Малышкин В.А. Применение методов эволюционной оптимизации для решения задач производственного планирования. Харьков, ХНУРЭ. 2011. С.97-102. 5. Мазур И., Шапиро В. Управление проектами. М.: ОМЕГА. 2007. 6. Будков В.М. Математические основы управления проектами. М.: Наука. 2005.

Поступила в редколлегию 11.12.2012

Гвоздинский Анатолий Николаевич, канд. тех. наук, профессор кафедры искусственного интеллекта ХНУРЭ. Научные интересы: оптимизация процедур принятия решений в сложных системах управления. Адрес: Украина, 61115, Харьков, ул. Академика Ляпунова, 7, кв. 9, тел. 702-38-23.

Янов Дмитрий Михайлович, студент, бакалавр специальности интеллектуальные системы принятия решений, факультет КН ХНУРЭ. Адрес: Украина, 61172, ул. Луи Пастера, 322, кв. 109, тел. 716-36-99, email: dmyto.yanov@gmail.com.

**МАТЕМАТИЧЕСКИЕ МОДЕЛИ ТЕХНОЛОГИЧЕСКОГО
ПРОЦЕССА НАПЫЛЕНИЯ НАНОСТРУКТУРИРОВАННЫХ
ДИЭЛЕКТРИЧЕСКИХ ПЛЕНОК, ПОЛУЧЕННЫХ МЕТОДОМ
ИОННО-ПЛАЗМЕННОГО РАСПЫЛЕНИЯ**

Приводятся математические модели процесса напыления наноструктурированных диэлектрических пленок, полученных ионно-плазменным методом распыления кремния в атмосфере кислорода, азота и их смесей. Определяются оптимальные управляющие воздействия параметров напыления, применение которых дает возможность получать качественные диэлектрические пленки с заданными параметрами. Приведенные оптимальные управляющие воздействия могут быть использованы для АСУТП получения пленок.

1. Анализ применения и методов получения наноструктурированных диэлектрических пленок

В последние годы бурно развиваются такие направления научных и технологических исследований как «нанонаука» и «нанотехнология». Эти направления рассматриваются как основа новой промышленной революции.

В этой связи тонкие пленки и в особенности наноструктурированные тонкие пленки играют очень важную роль в современной технике. Их значение в научно-техническом прогрессе чрезвычайно велико. Они используются в самых разнообразных областях науки и техники, например, в качестве защитных покрытий, для преобразования солнечной энергии в электрическую, в сверхпроводниковых приборах, в интегральной и функциональной микро- и нанoeлектронике, компьютерной технике, в медицине.

В то же время современные методы получения тонких пленок являются не только инструментом для их нанесения, но и методом, который позволяет формировать состав, структуру и свойства пленок, придавая им тем самым свойства, качественно отличающиеся от свойств исходного материала. В результате создается возможность целенаправленного формирования свойств пленок и придания им статуса полифункциональности.

Одним из хорошо изученных методов получения тонких пленок является ионное плазменное напыление, реализуемое в магнетронных системах и в пеннинговской газоразрядной камере, в которых применение магнитного поля позволяет понизить на порядок рабочее давление и обеспечить высокие скорости напыления пленок [1-4]. К недостаткам оборудования, реализующего эти методы получения тонких пленок, можно отнести отсутствие автоматического управления технологическим процессом, основой которого служит математическая модель процесса нанесения тонких пленок - его количественное описание.

Целью работы является построение математической модели процесса получения диэлектрических пленок SiO_2 , Si_3N_4 , $\text{Si}_x\text{O}_y\text{N}_z$, а также структур на их основе. Свойства этих пленок хорошо изучены и они широко применяются в полупроводниковой электронике [5-7].

2. Математическое моделирование процесса напыления диэлектрических пленок и определение оптимальных управляющих воздействий

Процесс осаждения диэлектрических пленок характеризуется следующими управляющими воздействиями: давлением рабочего газа P , температурой подложки T и током разряда I , определяющим скорость осаждения, т.е. производительность процесса. Магнитное поле остается постоянным на уровне, обеспечивающем работу распылительного элемента при минимальном давлении в камере.

Масс-спектрометрический анализ остаточного и рабочего состава газов показал, что наиболее существенно влияет на процесс осаждения и электрофизические свойства диэлектрических пленок присутствие паров воды. Поэтому основным возмущающим воздей-

ствием нами выбрано количество паров воды, содержащейся в остаточной атмосфере подколпачного устройства.

Качество диэлектрических пленок характеризуется тремя основными параметрами: тангенсом угла диэлектрических потерь, диэлектрической проницаемостью, электрической прочностью. В [8] установлено, что условия, соответствующие минимальному значению тангенса диэлектрических потерь ($\text{tg}\delta$), обеспечивают получение пленок с наилучшими параметрами. В связи с этим для управления процессом осаждения принят критерий – минимум $\text{tg}\delta$ полученных пленок при заданном возмущающем воздействии γ :

$$\frac{\min \varphi(I, P, T, \gamma)}{\gamma} = \gamma_0, \quad (1)$$

где $\varphi(I, P, T, \gamma)$ – зависимость $\text{tg}\delta$ от I, P, T, γ .

Для получения математической модели качества каждой пленки в явном виде нами проведен трехфакторный, трехуровневый эксперимент со случайным изменением четвертого фактора – возмущающего воздействия. В ходе эксперимента пленки осаждали при трех различных значениях управляющих воздействий и всех возможных комбинациях этих значений.

В целях получения в данном режиме осаждения пленок при возможно более равномерном распределении $\text{tg}\delta$, в допустимом диапазоне изменений γ от 1 до 3 условных единиц, для каждой из 27 комбинаций управляющих воздействий эксперимент повторялся 5-10 раз. В результате обработки 600- 700 экспериментальных значений по каждой диэлектрической пленке нами получены методом наименьших квадратов полиномиальные математические модели качества пленок:

$$\varphi_{\text{SiO}_2} = (24870P^2 + 65,6I^2 + 0,1039\gamma^2 - 1685P - 31,094I + 1,672\gamma - 0,05T + 7,789) \times 10^{-2}, \quad (2)$$

где P – изменяется в интервале от 1×10^{-3} до 5×10^{-3} мм рт.ст., I изменяется в интервале от 0,2 до 0,3 А, T – изменяется в интервале от 160 до 250 °С

$$\varphi_{\text{Si}_3\text{N}_4} = (171300P^2 + 32,81I^2 + 0,00018T^2 + 0,495\gamma^2 - 0,0607T - 12,8581 - 965\gamma + 0,365\gamma - 0,0036T\gamma - 2,427\gamma - 95P\gamma + 12,951) \times 10^{-2},$$

здесь P – изменяется в интервале от 0,133 до 0,665 Па, I – изменяется в интервале от 0,18 до 0,28 А, T – изменяется в интервале от 160 до 300 °С;

$$\varphi_{\text{Si}_x\text{O}_y\text{N}_z} = (4,05 - 0,61\tilde{P} - 0,512\tilde{T} + 0,6924\tilde{\gamma} + 0,23185\tilde{I} + 0,31667\tilde{P}^2 + 1,0972\tilde{T}^2 + 0,06944\tilde{\gamma}^2 + 0,7244\tilde{I}^2 + 0,0027778\tilde{P}\tilde{T} + 0,15681\tilde{P}\tilde{\gamma} + 0,04381\tilde{P}\tilde{I} + 0,0055556\tilde{T}\tilde{\gamma} + 0,46389\tilde{T}\tilde{I} - 0,04111\tilde{\gamma}\tilde{I}) \times 10^{-3},$$

где $\tilde{P} = P \times 10^{-3} - 3$, $\tilde{T} = T \times 10^{-2} - 1,8$, $\tilde{I} = I \times 10 - 2,5$, $\tilde{\gamma} = \gamma - 2$, P изменяется в интервале от 1×10^{-3} до 5×10^{-3} мм рт.ст., T изменяется в интервале от 180 до 200 °С, I изменяется в интервале от 0,2 до 0,3 А.

Полученные модели справедливы в интервале изменений γ в диапазоне от 1 до 3 условных единиц. При этом все значения управляющих воздействий не выходят из области допустимых значений.

В случае квадратичной зависимости для получения оптимальных управляющих воздействий достаточно применить принцип необходимых условий экстремума [9]:

$$\frac{d\varphi_{\text{SiO}_2}}{dP} = (2 \times 248700P - 183,5\gamma - 1686) \times 10^{-3} = 0, \quad (3)$$

$$\frac{d\varphi_{\text{SiO}_2}}{dI} = (2 \times 65,6I - 2,755\gamma - 31,094) \times 10^{-3} = 0.$$

Решение системы уравнений дает значения оптимальных управляющих воздействий при выращивании диэлектрической пленки SiO_2 :

$$P_{\text{opt}} = (2,818 + 0,2775\gamma) \times 10^{-3}, I_{\text{opt}} = 0,196 + 0,037\gamma, T_{\text{opt}} = 168,6 + 10\gamma. \quad (4)$$

Определение оптимальных управляющих воздействий при получении пленки Si_3N_4 осуществляется решением следующей системы уравнений:

$$\frac{d\varphi_{\text{Si}_3\text{N}_4}}{dP} = (2 \times 171 \times 300P - 965 - 95\gamma) \times 10^{-3} = 0, \quad (5)$$

$$\frac{d\varphi_{\text{Si}_3\text{N}_4}}{dI} = (2 \times 32,81 - 12,858 - 2,427\gamma) \times 10^{-3} = 0, \quad (6)$$

$$\frac{d\varphi_{\text{Si}_3\text{N}_4}}{dT} = (2 \times 0,00018T - 0,0607 - 0,0036\gamma) \times 10^{-3} = 0.$$

Решение системы уравнений (6) дает следующие значения оптимальных управляющих воздействий при выращивании диэлектрической пленки Si_3N_4 :

$$P_{\text{opt}} = (2,818 + 0,2775\gamma) \times 10^{-3}, I_{\text{opt}} = 0,196 + 0,037\gamma, T_{\text{opt}} = 168,6 + 10\gamma.$$

Для диэлектрической пленки $\text{Si}_x\text{O}_y\text{N}_z$ система уравнений для определения оптимальных управляющих воздействий будет следующей:

$$\frac{d\varphi_{\text{Si}_x\text{O}_y\text{N}_z}}{d\tilde{P}} = -0,61056 - 0,63334\tilde{P} + 0,0027778\tilde{T} - 0,1568\tilde{\gamma} + 0,4361\tilde{I} = 0,$$

$$\frac{d\varphi_{\text{Si}_x\text{O}_y\text{N}_z}}{d\tilde{I}} = 0,23185 + 1,4488\tilde{I} + 0,04361\tilde{P} + 0,46389\tilde{T} - 0,04111\tilde{\gamma} = 0, \quad (7)$$

$$\frac{d\varphi_{\text{Si}_x\text{O}_y\text{N}_z}}{d\tilde{T}} = -0,512 + 2,1944\tilde{T} + 0,027778\tilde{P} + 0,0055556\tilde{\gamma} + 0,46389\tilde{I} = 0.$$

После преобразования (7) и решения системы линейных уравнений с параметром $\tilde{\gamma}$ в правой части получим:

$$P_{\text{opt}} = 0,982 + 0,246\tilde{\gamma}, I_{\text{opt}} = 0,283 + 0,0235\tilde{\gamma}, \tilde{T}_{\text{opt}} = 0,2919 + 0,007802\tilde{\gamma}.$$

С учетом того, что $\tilde{P} = P \times 10^{-3} - 3$, $\tilde{T} = T \times 10^{-2} - 1,8$, $\tilde{I} = I \times 10 - 2,5$, $\tilde{\gamma} = \gamma - 2$, получим следующие оптимальные управляющие воздействия:

$$P_{\text{opt}} = (3,49 + 0,246\gamma) \times 10^{-3}, I_{\text{opt}} = 0,00235 + 0,217\gamma, T_{\text{opt}} = 210,75 - 0,7802\gamma.$$

При изменении γ от 1 до 3 условных единиц все значения оптимальных управляющих воздействий не выходят из области допустимых значений.

3. Выводы

В результате активного планируемого эксперимента получены математические выражения, позволяющие управлять процессом напыления тонких пленок. Определены оптимальные управляющие воздействия, при использовании которых обеспечивается получение диэлектрических пленок с заданными параметрами. На основе полученных математических выражений возможно создание автоматического управления процессом, что значительно повысит воспроизводимость параметров диэлектрических пленок.

Дальнейшей задачей является исследование эффективности использования оптимального управления, формирование алгоритма управления для АСУТП и построение функциональной схемы для его реализации.

Список литературы: 1. Петухов В.Ю., Гумаров Г.Г. Ионно-лучевые методы получения тонких пленок. Казань, 2010. 87 с. http://www.ksu.ru/f6/k5/bin_files/petukhov_ibm!33.pdf. 2. Ведерников В.А., Гурин В.Н., Данилина Т.И. Получение равномерных по толщине пленок при распылении в низковольтном пеннинговском разряде. ПТЭ, 1976, №3. С. 243-246. 3. Гурин В.Н. Влияние проникающей плазмы на свойства диэлектрика и границы раздела полупроводник-диэлектрик при реактивном катодном распылении // Радиоэлектроника и информатика. 2001. №4. С.29-32. 4. Берлин Е.В., Сейдман Л.А. Ионно-плазменные процессы в тонкопленочной технологии. М.: Техносфера, 2010. 528 с. 5. Гурин В.Н., Федоренко В.И., Сологуб В.Е. Влияние параметров диэлектрических пленок на стабильность МДП структур со сложным изолятором // Радиоэлектроника систем управления летательными аппаратами. Харьков, ХАИ, 1981. С.76-81. 6. Schwarts N. Preparation and properties of reactive sputtered Silicon nitride. 1963 Trans 10th Nat. Vacuum Sympos. Amer. Vacuum Soc. Oct., 1963. P.325-329. 7. Koski K. Properties of zirconium oxide thin films deposited by pulsed reactive magnetron sputtering / K. Koski, J. Holsa, P. Juliet // Surface and Coatings Technology. 1999. V.120. P.303-312. 8. Гурин В.Н. О возможности оптимизации технологического процесса получения диэлектрических пленок в низковольтном пеннинговском разряде // Получение и свойства тонких пленок. Киев. ИПМ АН УССР, 1981. Вып.7. С. 100-104. 9. Налимов В.В., Чернова М.А. Статистические методы планирования экстремальных экспериментов. М.: Наука, 1965. 340 С.

Поступила в редколлегию 07.12.2012

Гурин Валерий Николаевич, д-р техн. наук, профессор кафедры «Экономической кибернетики и управления экономической безопасностью» ХНУРЭ. Научные интересы: разработка инвестиционных проектов. Хобби: горный и водный туризм. Адрес: Украина, 31019, Харьков, ул. Тимирязева-28, кв.221, тел: моб. 0634935462, дом. 3764242, раб.7021490.

Фирсов Алексей Геннадиевич, канд. техн. наук, ассистент кафедры «Технологии автоматизации производства РЭА и ЭВС» ХНУРЭ. Научные интересы: разработка встраиваемых систем. Хобби: психология, философия. Адрес: Украина, 61099, Харьков, бул. Б. Хмельницкого, дом 10, кв. 69; моб. тел: 0501445197

Гурин Дмитрий Валерьевич, студент группы АКИТ- 10-2 ХНУРЭ. Научные интересы: исследование процессов автоматизации технологических процессов. Хобби: горный туризм, путешествия. Адрес: Украина, 31019, Харьков, ул. Тимирязева, 28, кв.221; тел: моб.0936455278, дом. 3764242.

УДК 004.896(06)

Е.С.САКАЛО

ИССЛЕДОВАНИЕ И ПОСТРОЕНИЕ ОБЛАЧНОЙ МОДЕЛИ УСЛУГ В CLOUD-СРЕДЕ

Предлагается публичная модель облака для поисковых систем с использованием распределённых вычислений для больших наборов данных.

1. Введение

Облачные вычисления представляют собой динамически масштабируемый способ доступа к внешним вычислительным ресурсам в виде сервиса, предоставляемого посредством Интернета, при этом пользователю не требуется никаких особых знаний об инфраструктуре “облака” или навыков управления этой “облачной” технологией.

Под “облаком”, подразумевается не сам Интернет, а весь тот набор аппаратного и программного обеспечения, который обеспечивает обработку и исполнение клиентских заявок “с той стороны”. Если не мудрить с определениями и вглядеться в суть, даже самое простое действие посредством Интернета – запрос страницы сайта в какой-то степени представляет собой пример облачного вычисления. Интернет, по сути, и возник именно для того, чтобы предоставлять удалённый доступ к различным вычислительным ресурсам.

Таким образом, распространившееся в последнее время мнение об облачных вычислениях как о чём-то специфическом, имеющем отношение к научным исследованиям или мощным корпоративным системам как минимум необъективно, а с точки зрения клиента – конечного пользователя, и вовсе не соответствует действительности.

Как результат, концепция Cloud Computing – облачных вычислений – позволяет организациям повысить эффективность предоставления услуг, упростить управление и улучшить согласование ИТ-сервисов с меняющимися потребностями бизнеса.

Список литературы: 1. Петухов В.Ю., Гумаров Г.Г. Ионно-лучевые методы получения тонких пленок. Казань, 2010. 87 с. http://www.ksu.ru/f6/k5/bin_files/petukhov_ibm!33.pdf. 2. Ведерников В.А., Гурин В.Н., Данилина Т.И. Получение равномерных по толщине пленок при распылении в низковольтном пеннинговском разряде. ПТЭ, 1976, №3. С. 243-246. 3. Гурин В.Н. Влияние проникающей плазмы на свойства диэлектрика и границы раздела полупроводник-диэлектрик при реактивном катодном распылении // Радиоэлектроника и информатика. 2001. №4. С.29-32. 4. Берлин Е.В., Сейдман Л.А. Ионно-плазменные процессы в тонкопленочной технологии. М.: Техносфера, 2010. 528 с. 5. Гурин В.Н., Федоренко В.И., Сологуб В.Е. Влияние параметров диэлектрических пленок на стабильность МДП структур со сложным изолятором // Радиоэлектроника систем управления летательными аппаратами. Харьков, ХАИ, 1981. С.76-81. 6. Schwarts N. Preparation and properties of reactive sputtered Silicon nitride. 1963 Trans 10th Nat. Vacuum Sympos. Amer. Vacuum Soc. Oct., 1963. P.325-329. 7. Koski K. Properties of zirconium oxide thin films deposited by pulsed reactive magnetron sputtering / K. Koski, J. Holsa, P. Juliet // Surface and Coatings Technology. 1999. V.120. P.303-312. 8. Гурин В.Н. О возможности оптимизации технологического процесса получения диэлектрических пленок в низковольтном пеннинговском разряде // Получение и свойства тонких пленок. Киев. ИПМ АН УССР, 1981. Вып.7. С. 100-104. 9. Налимов В.В., Чернова М.А. Статистические методы планирования экстремальных экспериментов. М.: Наука, 1965. 340 С.

Поступила в редколлегию 07.12.2012

Гурин Валерий Николаевич, д-р техн. наук, профессор кафедры «Экономической кибернетики и управления экономической безопасностью» ХНУРЭ. Научные интересы: разработка инвестиционных проектов. Хобби: горный и водный туризм. Адрес: Украина, 31019, Харьков, ул. Тимирязева-28, кв.221, тел: моб. 0634935462, дом. 3764242, раб.7021490.

Фирсов Алексей Геннадиевич, канд. техн. наук, ассистент кафедры «Технологии автоматизации производства РЭА и ЭВС» ХНУРЭ. Научные интересы: разработка встраиваемых систем. Хобби: психология, философия. Адрес: Украина, 61099, Харьков, бул. Б. Хмельницкого, дом 10, кв. 69; моб. тел: 0501445197

Гурин Дмитрий Валерьевич, студент группы АКИТ- 10-2 ХНУРЭ. Научные интересы: исследование процессов автоматизации технологических процессов. Хобби: горный туризм, путешествия. Адрес: Украина, 31019, Харьков, ул. Тимирязева, 28, кв.221; тел: моб.0936455278, дом. 3764242.

УДК 004.896(06)

Е.С.САКАЛО

ИССЛЕДОВАНИЕ И ПОСТРОЕНИЕ ОБЛАЧНОЙ МОДЕЛИ УСЛУГ В CLOUD-СРЕДЕ

Предлагается публичная модель облака для поисковых систем с использованием распределённых вычислений для больших наборов данных.

1. Введение

Облачные вычисления представляют собой динамически масштабируемый способ доступа к внешним вычислительным ресурсам в виде сервиса, предоставляемого посредством Интернета, при этом пользователю не требуется никаких особых знаний об инфраструктуре “облака” или навыков управления этой “облачной” технологией.

Под “облаком”, подразумевается не сам Интернет, а весь тот набор аппаратного и программного обеспечения, который обеспечивает обработку и исполнение клиентских заявок “с той стороны”. Если не мудрить с определениями и вглядеться в суть, даже самое простое действие посредством Интернета – запрос страницы сайта в какой-то степени представляет собой пример облачного вычисления. Интернет, по сути, и возник именно для того, чтобы предоставлять удалённый доступ к различным вычислительным ресурсам.

Таким образом, распространившееся в последнее время мнение об облачных вычислениях как о чём-то специфическом, имеющем отношение к научным исследованиям или мощным корпоративным системам как минимум необъективно, а с точки зрения клиента – конечного пользователя, и вовсе не соответствует действительности.

Как результат, концепция Cloud Computing – облачных вычислений – позволяет организациям повысить эффективность предоставления услуг, упростить управление и улучшить согласование ИТ-сервисов с меняющимися потребностями бизнеса.

В настоящее время применяются как публичные (public), так и частные (private) модели Cloud-среды.

Публичными моделями Cloud-решений может пользоваться любой человек с доступом в Интернет. Аспект аутсорсинга затрудняет поддержание целостности и конфиденциальности данных, обеспечение доступности данных и готовности сервиса, а также демонстрацию соответствия нормативным требованиям.

Фактически концепция Cloud Computing передает значительную часть контроля над данными и операциями от клиентской организации к ее поставщику Cloud-сервисов подобно тому, как организация доверяет часть своих ИТ-операций аутсорсинговой компании. Даже такие базовые задачи, как установка пакетов с исправлениями и конфигурирование межсетевых экранов, могут стать обязанностью поставщика Cloud-сервисов, а не конечного пользователя.

При применении модели доступа к ИТ-ресурсам Cloud Computing (облачные вычисления) информационные сервисы предоставляются таким образом, что обеспечивающие технологии становятся практически невидимыми за пределами пользователя устройства. А поскольку это позволяет отделить приложения и сервисы от обеспечивающей инфраструктуры и тем самым позволить бизнесу быстрее адаптироваться к изменениям, облачные вычисления могут быть частью стратегии по повышению динамичности предприятия.

При разработке модели поисковой системы в Cloud-среде за основу была взята поисковая система Amazon CloudSearch. Amazon CloudSearch представляет собой полностью управляемый поисковый сервис в облаке, позволяет заказчикам легко и быстро интегрировать масштабируемые функции поиска в своих приложениях.

Amazon CloudSearch легко масштабируется, когда объем поиска данных увеличивается, а разработчики могут изменить параметры поиска и применить новые настройки в любое время без необходимости загружать данные еще раз. Эта технология направлена на упрощение процесса для веб-разработчиков, желающих интегрировать поиск в своих размещенных сайтах.

Поиск домена для каждого набора данных, которые необходимо сделать доступными, имеет один или несколько случаев, каждый из которых обладает конечным объемом оперативной памяти и ресурсов процессора для индексирования данных и обработки запросов. Количество поисковых случаев домена зависит от количества документов, а также объема и сложности запросов. В управляемых услугах поисковой системы в облачной среде система определяет размер и количество поисковых случаев, необходимых для достижения низкой задержки, высокой производительности поиска пропускной способности. Когда пользователь загружает данные и настройки индексов, система поиска строит индекс и выбирает соответствующий начальный экземпляр типа поиска, чтобы индекс клиента мог храниться в оперативной памяти.

Поисковая система будет масштабироваться до поиска области (или раздела индекса по нескольким экземплярам, если клиент уже в самом большом). И наоборот, когда объем данных сокращается, система масштабируется до поиска меньших случаев или меньшего типа поиска, например, если индекс помещается на одном разделе.

Задачи для достижения цели: 1) Детально проанализировать работу нескольких поисковых сред [1, 5]. 2) Построить архитектуру программы, основанную на облачных вычислениях [1, 6, 7].

В качестве объекта разработки выступает поисковая система в Cloud-среде. С учетом всех недостатков существующих поисковых систем возникает острая задача поиска алгоритма для построения поисковой системы на основе облачных вычислений.

2. Внедрение облачных вычислений при проектировании настройки облака

Для решения задачи внедрения облачных вычислений при проектировании настройки облака в работе использовалась технология Map Reduce. В рамках этой технологии Google была предложена абстрактная модель, которая позволила выражать простые вычисления, в то же время пряча сложные детали параллелизации: обработку ошибок, распределение данных, балансировку нагрузки.

Первая реализация этой модели была выполнена на основе распределенной файловой системы той же компании GFS (Google File System). Эта реализация активно используется в программных продуктах самой Google, но является сугубо проприетарной и недоступна для использования вне Google.

Альтернативная, свободно доступная реализация Hadoop MapReduce (с открытыми исходными текстами) была выполнена в проекте Hadoop сообщества Apache. Архитектура облачной модели основана на доминировании веб-приложений над обычными функциями ОС. Ключевая роль при этом отводится браузеру. Стратегия создания нового продукта подразумевает архитектуру, нетребовательную к аппаратным ресурсам персонального компьютера, используемого для выхода в сеть Интернет. Эта тенденция переноса центра тяжести с пользователя ПК на Интернет-ресурсы и соответствует идеологии облачных вычислений. В этой модели вычисления производятся над множествами входных пар “ключ-значение”, и в результате каждого вычисления также производится некоторое множество результирующих пар “ключ-значение”. Для представления вычислений в среде MapReduce используются две основные функции: Map и Reduce. Обе функции явно кодируются разработчиками приложений в среде MapReduce.

Функция Map в цикле обрабатывает каждую пару из множества входных пар и производит множество промежуточных пар “ключ-значение”. Среда MapReduce группирует все промежуточные значения с одним и тем же ключом I и передает их функции Reduce.

Функция Reduce получает значение ключа I и множество значений, связанных с этим ключом. В типичных ситуациях каждая группа обрабатывается (в цикле) таким образом, что в результате одного вызова функции образуется не более одного результирующего значения.

Реализации MapReduce от Google и Hadoop ориентированы на использование в кластерной распределенной среде со следующими основными характеристиками:

- узлы среды выполнения MR-приложений обычно представляют собой компьютеры общего назначения с операционной системой Linux;
- используется стандартное сетевое оборудование с адаптерами, рассчитанными на скорости передачи в 100 мегабит в секунду или 1 гигабит в секунду, но средняя пропускная способность существенно ниже;
- кластер состоит из сотен или тысяч машин, так что вполне вероятны отказы отдельных узлов;
- для хранения данных используются недорогие дисковые устройства, подключенные напрямую к отдельным машинам;
- для управления данными, хранящимися на этих дисках, применяется распределенная файловая система;
- пользователи представляют свои задания в систему планирования; каждое задание состоит из некоторого набора задач, которые отображаются планировщиком на набор узлов кластера. Рассмотрим реализацию модели MapReduce в распределенной среде.

Вызовы Map распределяются по нескольким узлам кластера путем разделения входных данных на M непересекающихся групп (split). Входные группы могут параллельно обрабатываться на разных машинах. Вызовы Reduce распределяются путем разделения пространства промежуточных ключей на R частей с использованием некоторой функции разделения (например, функции хэширования). Число разделов R и функция разделения задаются пользователем.

Выполнение MR-программы происходит следующим образом. Сначала среда MapReduce расщепляет входной файл на M частей, размер которых может задаваться пользователем. Затем сразу в нескольких узлах кластера запускается основная программа MapReduce. Один из ее экземпляров играет специальную роль и называется распорядителем (master). Остальные экземпляры являются исполнителями (worker), которым распорядитель назначает работу. Распорядитель должен назначить исполнителям для выполнения M задач Map и R задач Reduce. В библиотеке MapReduce поддерживается возможность чтения входных данных в нескольких разных форматах. Например, в режиме “text” каждая строка трактуется как пара “ключ-значение”, где ключ – это смещение до данной строки от начала файла, а значение – содержимое строки. В другом распространенном формате входные данные представляются в виде пар “ключ-значение”, отсортированных по значениям ключа. В каждой реализации формата входных данных известно, каким образом следует расщеплять данные на осмысленные части, которые обрабатываются отдельными задачами Map (например, данные формата “text” расщепляются только по границами строк).

Пользователи могут добавить к реализации собственные форматы входных данных, обеспечив новую реализацию интерфейса reader (в реализации Hadoop – RecordReader). Reader не обязательно должен читать данные из файла, можно легко определить reader, читающий данные из базы данных или из некоторой структуры в виртуальной памяти.

Аналогичным образом поддерживаются возможности генерации данных в разных форматах, и имеется простая возможность определения новых форматов результирующих данных.

Основными особенностями системы есть: скорость загрузки, выхода в Интернет, получения электронной почты, тесная интеграция с интернет-сервисами, надежность работы, обеспечение безопасности в автоматическом режиме, автоматическое обновление, простота. Как и объем данных, система автоматически масштабирует поиск домена в соответствии с требованиями.

Ключевым элементом платформы есть средства автоматического обнаружения, способные выделить компоненты сервиса и их изменения, включая виртуальные и физические серверы, сетевые соединения и приложения.

Каталог сервисов предоставляет меню облачных сервисов и должен сопровождаться реализацией специального пользовательского портала самообслуживания.

Вычислительные облака состоят из тысяч серверов, размещенных в датацентрах, обеспечивающих работу десятков тысяч приложений, которые одновременно применяют миллионы пользователей. Непременным условием эффективного управления такой крупномасштабной инфраструктурой является максимально полная автоматизация. Кроме того, для обеспечения различных видов пользователей - облачных операторов, сервис-провайдеров, посредников, ИТ-администраторов, пользователей приложений - защищенного доступа к вычислительным ресурсам облачная инфраструктура должна предусматривать возможность самоуправления и делегирования полномочий.

Концепция облачных вычислений значительно изменила традиционный подход к доставке, управлению и интеграции приложений. По сравнению с традиционным подходом, облачные вычисления позволяют управлять большими инфраструктурами, обслуживать различные группы пользователей в пределах одного облака.

3. Заключение

Предложена облачная модель услуг. Это гибкая, экономически эффективная и проверенная технология для доставки ИТ-сервисов организациям или индивидуальным потребителям с использованием сети Интернет. Ресурсы Cloud-среды быстро разворачиваются и легко масштабируются, при этом инициализация всех процессов, приложений и сервисов осуществляется по требованию, независимо от местоположения потребителя или используемого компьютерного устройства. Модель с применением облачных вычислений является динамично масштабируемой для доступа к внешним вычислительным ресурсам в виде сервиса, предоставляемого посредством Интернета, при этом пользователю не требуется никаких особых знаний об инфраструктуре «облака» или навыков управления этой «облачной» технологией.

Практическая значимость. Данная модель – это единая архитектура, которая объединяет возможности по использованию «частных», «гибридных» и «публичных» облаков, а также позволяет совмещать их с традиционной ИТ-средой. Благодаря применению стандартных модульных компонентов, а также использованию разработанного метода позволяет оперативно находить необходимую информацию.

Список литературы: 1. *A distributed storage system for structured data* / Chang, J. Dean. Seattle, USA, 2006. С. 205-218. 2. Dean, J. and Ghemawat, S. MapReduce: Simplified data processing on large clusters. San Francisco, USA, 2004. С. 137-150. 3. Barroso, L. A., Dean, J., and Urs Hylzle, U. Web search for a planet: The Google cluster architecture. Seattle, USA, 2003. С. 22–28. 4. *Hadoop Map/Reduce Tutorial* [Электронный ресурс]. Электрон. дан. Режим доступа: http://hadoop.apache.org/common/docs/current/mapred_tutorial.html, свободный. 5. CloudEra [Электронный ресурс] Электрон. дан. Режим доступа: <http://www.cloudera.com/>, свободный. 6. How to Create and Debug an Amazon Elastic MapReduce Job Flow, [www/ URL: http://aws.amazon.com/articles/3938](http://aws.amazon.com/articles/3938) 7. Екатерина Баранова, «Концепция Cloud computing» www.itcontent.ru/archives/blog/cloud_computing.

Поступила в редколлегию 18.11.2013

Сакало Евгений Сергеевич, канд. техн. наук, доцент кафедры информатики ХНУРЭ. Научные интересы: облачные технологии, проектирование облаков различной сложности. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 0577021419. Email: yevgen.sakalo@gmail.com.

ИНФОРМАЦИОННЫЕ МОДЕЛИ ДАННЫХ И КРИТЕРИИ ОЦЕНКИ КАЧЕСТВА КОМПОНОВКИ СТАТЕЙ НА ПОЛОСЕ ПРИ ДОПЕЧАТНОЙ ПОДГОТОВКЕ ИЗДАНИЙ

На основе исследования процесса допечатной подготовки издания ставится информационно-технологическая задача формирования оригинал-макета издания. В рамках решения этой задачи проводится унификация данных и разрабатываются модели их унифицированных структур, а также строятся два взаимосвязанных набора критериев для оценки качества компоновок статей на полосах издания и компоновок их фрагментов.

Введение

Тенденции развития мирового рынка полиграфических товаров и услуг убеждают в необходимости и эффективности внедрения новейших информационных технологий в целях улучшения технико-экономических показателей издательских проектов. Весь процесс изготовления печатного издания разделяют на три стадии, первая из которых – допечатная подготовка, в значительной мере определяющая качество и себестоимость издания.

Доминирующей частью допечатной подготовки издания является редакционный этап (РЭДПИ), характеризующийся разнородностью применяемых издательских концепций, информационных технологий, технических и программных средств. На основе анализа различных технологических процессов допечатной подготовки и личного опыта работы в издательстве создана информационная модель обобщенного РЭДПИ [1], охватывающая четыре технологических передела: 1) перенос авторских рукописей статей на электронные носители информации редакции; 2) корректировка текстовых и графических фрагментов статей; 3) формирование структур полос издания; 4) верстка издания.

Эта модель использована для постановки новой информационно-технологической задачи формирования оригинал-макета издания (ФОМИ), которая, в свою очередь, использована при постановке оптимизационной задачи размещения статей на полосе издания (G-задачи). Поставленная G-задача охватывает две взаимосвязанные подзадачи ФОМИ — «сборка фрагментов в статью» и «размещение статей на полосе издания». С математической точки зрения обе эти подзадачи относятся к классу оптимизационных задач размещения геометрических объектов, которые в наиболее общем представлении сводятся к поиску экстремума функционала F на некотором множестве расстановок геометрических объектов K , осуществляемых в области размещения Z ($Z \subset R$, где R – метрическое пространство) при выполнении множества ограничений D .

Разработанные методы и алгоритмы решения G-задачи [2] реализованы в автоматизированной системе допечатной подготовки изданий G1.

Цель исследования — создание методов и инструментальных средств допечатной подготовки изданий. *Задачи исследования* – разработка моделей унифицированных структур данных и построение наборов критериев для оценки качества компоновок статей на полосах издания. Эти модели данных и критерии оценки являются основным содержанием последующих разделов статьи.

1. Выдержки из постановки задачи формирования оригинал-макета издания

В кратком изложении предметная область Y задачи ФОМИ определена тройкой множеств $Y = (\Pi, \Omega, \Theta)$, где $\Pi = \{\pi\}$ – множество предметов; $\Omega = \{\omega\}$ – множество операций, заданных на Π ; $\Theta = \{\theta\}$ – множество отношений, заданных на Π и Θ .

Определены следующие типы предметов: издание, полоса издания, статья, фрагмент статьи, портфель, компоновка, оригинал-макет издания. Предмет каждого типа характеризуется множеством свойств и состояний.

Конечным продуктом, получаемым в процессе редакционно-издательской деятельности, является «издание». Издание (I) состоит из одной или нескольких «статей».

Издание можно рассматривать как совокупность «полос издания». Полосы издания составляют множество: $B = \{\beta_i\}, i = \overline{1, m}; m \in N$, где N – множество натуральных чисел. Различаются следующие части полосы издания: рабочая область (прямоугольной формы) и поля. В рабочей области полосы издания размещаются статьи.

Концептуально, наиболее важным предметом является статья. Статьи издания составляют множество: $A = \{\alpha_i\}, i = \overline{1, n}; n \in N$. Каждая статья характеризуется множеством свойств, разделяемых на три категории: идентификационные, геометрические и редакционно-издательские. Различаются следующие части статьи: базовая точка и граничная точка. Статья имеет прямоугольную форму и подлежит размещению в рабочей области полосы издания.

На основе статьи порождается множество Φ , состоящее из одного или нескольких элементов, которые названы «фрагментами статьи»: $\alpha_i \Rightarrow \Phi_i = \{\phi_{ij}\}$, где $j = \overline{1, k}; k \in N$. С текстографической точки зрения фрагмент является текстом или изображением. Различаются три типа фрагментов: текстовый, «текст-вставка», графический. Каждый фрагмент характеризуется множеством свойств, разделяемых (как и для статьи) на три категории: идентификационные, геометрические и редакционно-издательские. Различаются следующие части фрагмента: базовая точка, граничная точка. Фрагмент имеет прямоугольную форму и располагается в границах статьи.

Данные о статьях и их фрагментах хранятся в «портфелях». Портфели составляют множество $H = H_\alpha \cup H_\phi$, где H_α – множество портфелей статей полос издания, H_ϕ – множество портфелей фрагментов.

Компоновка представляет собой или совокупность фрагментов, размещенных в рабочей области статьи, или совокупность статей, размещенных в рабочей области полосы издания. Компоновки составляют множество $K = K_\phi \cup K_\alpha$, где K_ϕ – множество компоновок фрагментов, K_α – множество компоновок статей. Главными свойствами компоновки являются форма (прямоугольник), линейные размеры (ширина, высота) и координаты базовых точек размещенных объектов (фрагментов или статей).

Конечным результатом, получаемым в рамках РЭДПИ, является «оригинал-макет».

Оригинал-макеты составляет множество: $M = \{M_0, M_1, M_2, M_3\}$, где M_0 – подмножество промежуточных нецветоделенных оригинал-макетов без спуска полос, M_1 – окончательный нецветоделенный оригинал-макет без спуска полос, M_2 – окончательный нецветоделенный оригинал-макет со спуском полос, M_3 – подмножество окончательных цветоделенных оригинал-макетов со спуском полос.

2. Унификация и модели данных к задаче формирования оригинал-макета издания

Указанные и другие свойства введенных в рассмотрение предметов задают совокупность данных. При решении задачи ФОМИ эта совокупность изменяется на каждом шаге решения до тех пор, пока не будут получены необходимые значения атрибутов, характеризующих эти данные. Для успешного решения указанной задачи требуется детальное описание и структурирование данных, унификация структур данных, а также выбор формы и способа представления унифицированных структур данных в базе данных издания. Представим наиболее важную структуру данных.

Идентификационные атрибуты данных, характеризующие предметы, позволяют различать сами предметы и их части. Идентификаторами предметов приняты: дата (выпуска издания), номер (издания, полосы издания), код (статьи, фрагмента статьи, автора статьи, компоновки, портфеля, оригинал-макета), название (издания, статьи, фрагмента). Геометрические атрибуты определяют форму предмета или его части (статьи, фрагмента, рабочей области полосы издания), а также месторасположение предмета (статьи, фрагмента) в области размещения. Редакционно-издательские атрибуты определяют широкий диапазон разнородных свойства предметов, которые необходимо учитывать при допечатной подго-

товке издания (от количества страниц издания и качества компоновки до используемых шрифтов, количества колонок и цветности статей).

Для предмета каждого типа введен унифицированный список L его атрибутов: $L = (L_I, L_G, L_P)$, где L_I, L_G, L_P — подсписки, соответственно, идентификационных, геометрических и редакционно-издательских атрибутов. Детализируем эти подсписки для предмета типа «статья»: $L^A = (L_I^A, L_G^A, L_P^A)$. $L_I^A = (\text{namepub}, \text{numpub}, \text{numbar}, \text{author}, \text{codeart}, \text{titleart})$, где namepub — атрибут, определяющий название издания, к которому принадлежит статья; numpub — сквозной номер издания, которому принадлежит статья; numbar — номер полосы издания, к которой приписана статья; author — идентификатор автора статьи; codeart и titleart определяют код и название статьи. $L_G^A = (\text{form}, \text{sizes}, \text{coord}, \text{square})$, где form — тип геометрической формы статьи. В соответствии с редакционно-издательской практикой в качестве типа геометрической формы выбран прямоугольник; sizes — вектор линейных размеров статьи после сборки фрагментов ($\text{sizes} = (\text{width}, \text{height})$), где width и height — ширина и высота статьи; coord — вектор координат базовой точки статьи ($\text{coord} = (x, y)$), где x и y — координаты x и y базовой точки статьи (в прямоугольной декартовой системе координат); square — площадь статьи после сборки фрагментов. $L_P^A = (\text{status}, \text{recsym}, \text{recsq}, \text{equsym}, \text{imppub}, \text{impread}, \text{adv}, \text{topic}, \text{type}, \text{frag}, \text{cost}, \text{fee}, \text{range}, \text{color}, \text{col})$, где status — атрибут, определяющий состояние (степень подготовленности) статьи; recsym — заказанное автору количество знаков; recsq — эквивалентная заказанная площадь статьи; equsym — эквивалент площади статьи в знаках после сборки фрагментов; imppub и impread — издательская и читательская информационная важность статьи; adv — индикатор того, является ли статья явной рекламой; topic — массив, каждый элемент topic (*) которого — идентификатор тем тематически связанных статей; type — структурный (текстографический) тип статьи; frag — количество фрагментов в статье; cost — стоимостные затраты на подготовку статьи на первом и втором технологических переделах; fee — размер гонорара автору за написание статьи или издателю за ее опубликование; range — ранг статьи (в портфеле статей); color — цветность (количество цветов при печати) статьи; col — количество колонок в статье. Все атрибуты списка L^A объединяются в агрегат STRARTICLE.

Все атрибуты и их описания, дополненные указаниями о типе организации, сфере действия и другими характеристиками (таблица) составляют одну из объектно-ориентированных моделей унифицированных структур данных для задачи ФОМИ. Подчеркнем, что эти модели в совокупности важны не столько для разработки оптимизационных методов размещения статей на полосе издания, сколько для проектирования автоматизированной системы допечатной подготовки издания, претендующей на реальное применение в повседневной работе издательств.

Условные обозначения в таблице: в столбце 4 «S» — одиночный, «D» — массив; 5 — «I» — входной, «O» — выходной, «W» — внутренний; 6 — «1» — постоянный, «2» — условно-постоянный, «3» — переменный; 7 — «C» — символьный, «A» — арифметический, «R» — запись.

На рисунке отображены основные структуры данных с иллюстративными значениями геометрических атрибутов и главные связи этих структур.

3. Наборы критериев и вычисление показателей качества компоновок

В поставленной G-задаче для оценки компоновки фрагментов вводится функционал

$F_\Phi = \sum_i q_i^{(\Phi)} t_i^{(\Phi)}$, а для оценки компоновки статей — функционал $F_\alpha = \sum_j q_j^{(\alpha)} t_j^{(\alpha)}$, где $t_i^{(\Phi)}$ и $t_j^{(\alpha)}$ — одиночные показатели качества, соответственно, компоновки фрагментов и компоновки статей; $q_i^{(\Phi)}$ и $q_j^{(\alpha)}$ — весовые коэффициенты. Конкретизация этих функционалов путем уточнения набора показателей качества компоновок, уточнения их смысла и процедур вычисления значений показателей является неотъемлемой частью решения подзадач «сборка фрагментов в статью» и «размещение статей на полосе издания».

Таблица. Структура данных для предмета «статья»

Под- спи- сок	Имя со- ставного атрибута	Имя простого атрибута	Тип ор- ганизаци- и	Сфера действия	Степень измен- чивости	Тип зна- че- ния	Единица величи- ны	Пе- ре- дел
1	2	3	4	5	6	7	8	9
L_1^A		namepub	S	I	1	C	-	1
		numpub	S	I	1	C	-	1
		numbar	S	I	2	A	-	1
		author	S	I	1	C	-	1
		codeart	S	I	1	C	-	1
		title	S	I	1	C	-	1
L_G^A	sizes	form	S	I	1	C	-	1
		width	S	O	3	A	мм	3
		height	S	O	3	A	мм	3
	coord	x	S	O	3	A	мм	3
		y	S	O	3	A	мм	3
		square	S	W	3	A	мм ²	3
L_P^A		status	S	W	3	A	-	1
		recsym	S	I	1	A	зн.	1
		recsq	S	I	1	A	мм ²	1
		equesym	S	W	3	A	зн.	3
		imppub	S	W	2	A	-	1
		impread	S	W	2	A	-	1
		adv	S	W	1	C	-	1
		topic	D	I	2	C	-	2
		type	S	W	1	C	-	2
		frag	S	W	2	A	-	2
		cost	S	W	3	A	евро	2
		fee	S	I	1	A	евро	1
		range	S	W	3	A	-	3
		color	S	I	2	C	-	2
		col	S	O	2	A	-	4

При анализе публикаций не выявлено такого набора критериев, которые можно было бы эффективно применить для оценки качества компоновок (статей и фрагментов), что обусловило необходимость разработки двух наборов критериев, рассматриваемых далее.

Основываясь на геометрическом сходстве статей и фрагментов, введем два взаимосвязанных набора унифицированных критериев: один для оценки качества компоновки статей, другой — для оценки качества компоновки фрагментов. В каждом наборе критерии подразделены на две группы («экономические» и «эстетические»), которые соответствуют таким распространенным глобальным целям издательства как доход и эстетическая привлекательность издания (которая в определенной мере обуславливает престиж издания). Каждому критерию поставлен в соответствие показатель.

Для оценки качества компоновки статей введем набор из 14 скалярных критериев.

Экономические критерии

1. «Массовость». Показатель $t_1^{(\alpha)}$ — количество статей, размещенных на полосе издания. Значение $t_1^{(\alpha)}$ совпадает со значением счетчика размещенных статей.

2. «Покрытость». Показатель $t_2^{(\alpha)}$ — суммарная площадь, занятая статьями на полосе издания:

$$t_2^{(\alpha)} = \sum_{i=1}^{t_1^{(\alpha)}} s_i, \text{ где } s_i —$$

площадь размещенной i -й статьи.

3. «Заполненность». Показатель $t_3^{(\alpha)}$ — степень заполнения рабочей области полосы издания:

$$t_3^{(\alpha)} = \frac{t_2^{(\alpha)}}{S_c} 100\%,$$

где S_c — площадь рабочей области полосы издания.

4. «Значимость». Показатель $t_4^{(\alpha)}$ — суммарная информационная важность размещенных статей:

$$t_4^{(\alpha)} = \sum_{i=1}^{t_1^{(\alpha)}} v_i^{(1)} + \sum_{i=1}^{t_1^{(\alpha)}} v_i^{(2)}, \text{ где}$$

$v_i^{(1)}$ — издательская важность i -й статьи; $v_i^{(2)}$ — читательская важность i -й статьи (см. атрибуты $imprub$ и $impread$).

5. «Рейтинговость». Показатель $t_5^{(\alpha)}$ — суммарный рейтинг размещенных статей;

$$t_5^{(\alpha)} = \sum_{i=1}^{t_1^{(\alpha)}} (\hat{q}_1 s_i + \hat{q}_2 (v_i^{(1)} + v_i^{(2)})), \text{ где } \hat{q}_1 \text{ и } \hat{q}_2 — \text{положительные весовые коэффициенты.}$$

6. «Доходность». Показатель $t_6^{(\alpha)}$ — суммарный доход от размещенных статей;

$$t_6^{(\alpha)} = \sum_{i=1}^{t_1^{(\alpha)}} (v_i^{(3)} + v_i^{(4)}), \text{ где } v_i^{(3)} — \text{размер гонорара за написание } i\text{-й статьи; } v_i^{(4)} — \text{затраты на подготовку } i\text{-й статьи (см. атрибуты } fee \text{ и } cost).$$

Эстетические критерии

7. «Рекламность». Показатель $t_7^{(\alpha)}$ — степень заполнения площади полосы издания

$$\text{рекламными статьями: } t_7^{(\alpha)} = \frac{\sum_{i=1}^{t_1^{(\alpha)}} v_i^{(5)}}{t_2^{(\alpha)}} 100\%, \text{ где } v_i^{(5)} — \text{площадь } i\text{-й статьи, являющейся явной рекламой (см. атрибут } adv).$$

ИМЯ АГРЕГАТА
ТИП ПРЕДМЕТА

STRPUBLISHING
ИЗДАНИЕ

STRBAR
ПОЛОСА ИЗДАНИЯ

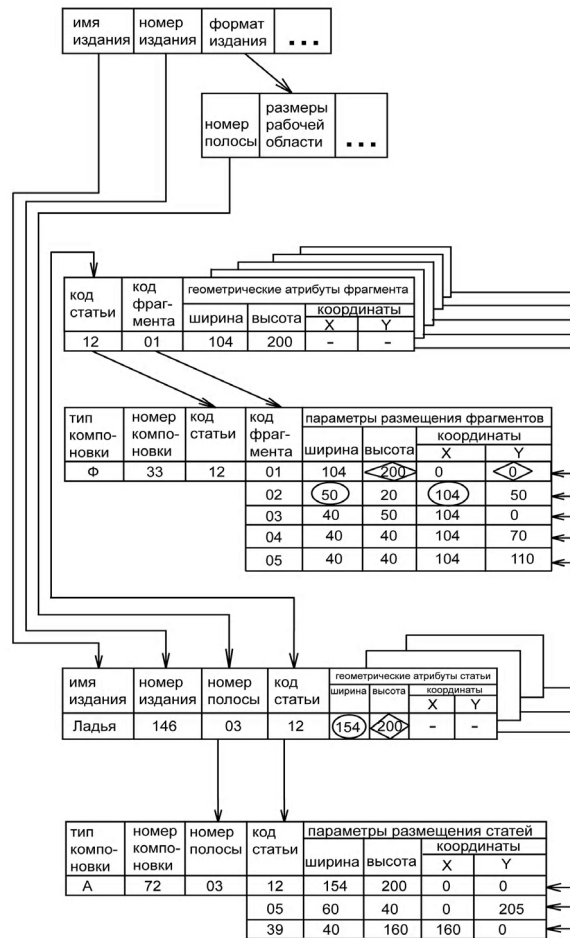
STRFRAGMENT
ФРАГМЕНТ

STRPLACEMENT
КОМПОНОВКА
(фрагментов)

STRARTICLE
СТАТЬЯ

STRPLACEMENT
КОМПОНОВКА
(статей)

АТРИБУТЫ



Основные связи структур данных в задаче формирования оригинал-макета издания

8. «Текстографичность». Показатель $t_8^{(\alpha)}$ — соотношение текстовых и графических фрагментов на полосе издания: $t_8^{(\alpha)} = \frac{v^{(6)}}{v^{(7)}} 100\%$, где $v^{(6)}$ — суммарное количество

текстовых фрагментов на полосе издания; $v^{(7)}$ — суммарное количество графических фрагментов.

9. «Центрированность». Показатель $t_9^{(\alpha)}$ — соотношение статей, расположенных в центральной зоне полосы издания и в периферийной зоне: $t_9^{(\alpha)} = \frac{v^{(8)}}{v^{(9)}} 100\%$, где $v^{(8)}$ и $v^{(9)}$

— количество статей, полностью расположенных, соответственно, в центральной и в периферийной зоне. (Замечание: если статья расположена одновременно и в центральной, и в периферийной зоне, то при вычислении показателя относим ее к той зоне, в которой находится большая часть площади статьи. Отметим, что соотношение между центральной и периферийной зонами может быть выбрано, например, в соответствии с правилом «золотого сечения», часто используемым в дизайнерской практике.)

10. «Сбалансированность». Показатель $t_{10}^{(\alpha)}$ — признак равномерности распределения статей в четырех четвертях рабочей области полосы издания:

$$t_{10}^{(\alpha)} = \begin{cases} 1, & \text{если } \max \left\{ \frac{S_1}{t_2^{(\alpha)}} 100\%, \frac{S_2}{t_2^{(\alpha)}} 100\%, \frac{S_3}{t_2^{(\alpha)}} 100\%, \frac{S_4}{t_2^{(\alpha)}} 100\% \right\} - \\ & - \min \left\{ \frac{S_1}{t_2^{(\alpha)}} 100\%, \frac{S_2}{t_2^{(\alpha)}} 100\%, \frac{S_3}{t_2^{(\alpha)}} 100\%, \frac{S_4}{t_2^{(\alpha)}} 100\% \right\} \leq \hat{d}; \\ 0, & \text{в противном случае} \end{cases}$$

здесь S_1, S_2, S_3, S_4 — суммарные площади, занятые статьями, соответственно, в 1-й, 2-й, 3-й, 4-й четвертях полосы издания; $\hat{d}$ — коэффициент равномерности, задаваемый экспертом. Равномерное распределение статей в четвертях рабочей области полосы соответствует равенству $t_{10}^{(\alpha)} = 1$.

11. «Линейная пропорциональность». Показатель $t_{11}^{(\alpha)}$ — количество нарушений линейной пропорциональности среди размещенных статей. Значение $t_{11}^{(\alpha)}$ совпадает со значением счетчика нарушений. (Замечание: условие нарушения пропорциональности линейных

размеров размещенной i -й статьи задается так: $\left(\frac{w_i}{h_i} > \check{d} \right) \vee \left(\frac{h_i}{w_i} > \check{d} \right)$, где w_i и h_i —

соответственно, ширина и высота размещенной i -й статьи; $\check{d}$ — коэффициент пропорциональности линейных размеров, задаваемый экспертом.)

12. «Пропорциональность площадей». Показатель $t_{12}^{(\alpha)}$ — признак нарушения пропорци-

ональности площадей статей, размещенных на полосе:
$$t_{12}^{(\alpha)} = \begin{cases} 1, & \text{если } \frac{s_i}{s_j} > \bar{d}; \\ 0, & \text{в противном случае,} \end{cases}$$

где s_i и s_j — соответственно площади размещенных i -й и j -й статей; $\bar{d}$ — коэффициент пропорциональности площадей, задаваемый экспертом. Нарушение пропорциональности площадей статей соответствует равенству $t_{12}^{(\alpha)} = 1$.

13. «Цветность». Показатель $t_{13}^{(\alpha)}$ — соотношение суммарных площадей цветных статей и черно-белых статей: $t_{13}^{(\alpha)} = \frac{s_p}{s_b} 100\%$, где s_p и s_b — суммарные площади, соответственно, цветных и черно-белых статей.

14. «Симметричность». Показатель: $t_{14}^{(\alpha)}$ — соотношение суммарных площадей статей, размещенных на левой и правой половинах рабочей области полосы издания:

$t_{14}^{(\alpha)} = \frac{S_l}{S_r} 100\%$, где S_l и S_r — суммарные площади статей, размещенных, соответствен-

но, в левой и правой половинах рабочей области полосы издания.

Для оценки качества компоновки фрагментов введем набор из 4 скалярных критериев.

Экономический критерий

1. «Заполненность». Показатель $t_1^{(\varphi)}$ — степень заполнения области размещения фрагментов: $t_1^{(\varphi)} = \frac{S^{(\varphi)}}{S^{(\alpha)}} 100\%$, где $S^{(\varphi)}$ — суммарная площадь, занятая фрагментами; $S^{(\alpha)}$ — площадь области размещения фрагментов статьи.

Эстетические критерии

2. «Линейная пропорциональность». Показатель $t_2^{(\varphi)}$ — количество нарушений линейной пропорциональности среди размещенных фрагментов. Значение $t_2^{(\varphi)}$ совпадает со значением счетчика нарушений. (Замечание: условие нарушения пропорциональности линейных размеров размещенного i -го фрагмента зададим также, как и для статей:

$\left(\frac{w_i}{h_i} > \bar{d}\right) \vee \left(\frac{h_i}{w_i} > \bar{d}\right)$, где w_i и h_i — соответственно, ширина и высота размещенного i -го фрагмента; $\bar{d}$ — коэффициент линейной пропорциональности, задаваемый экспертом.)

3. «Пропорциональность площадей». Показатель $t_3^{(\varphi)}$ — признак нарушения пропорциональности площадей размещенных фрагментов:
$$t_3^{(\varphi)} = \begin{cases} 1, & \text{если } \frac{s_i}{s_j} > \bar{d}; \\ 0, & \text{в противном случае} \end{cases}, \text{ где } s_i$$

и s_j — соответственно площади размещенных i -го и j -го фрагментов. Нарушение пропорциональности площадей фрагментов соответствует равенству $t_3^{(\varphi)} = 1$.

4. «Сегментированность текстового фрагмента». Показатель $t_4^{(\varphi)}$ — степень дробления

текстового фрагмента: $t_4^{(\varphi)} = \frac{v^{(10)}}{v^{(11)}} 100\%$, где $v^{(10)}$ — количество сегментов текстового

фрагмента; $v^{(11)}$ — максимально допустимое количество сегментов текстового фрагмента, задаваемое экспертом.

Обобщенные показатели качества компоновки статей ($T^{(\alpha)}$) и компоновки фрагментов

($T^{(\varphi)}$) вычисляются как взвешенные суммы показателей: $T^{(\alpha)} = \sum_{i=1}^{14} q_i^{(\alpha)} t_i^{(\alpha)}$,

$T^{(\varphi)} = \sum_{j=1}^4 q_j^{(\varphi)} t_j^{(\varphi)}$, где $q_i^{(\alpha)}$, $q_j^{(\varphi)}$ — положительные весовые коэффициенты.

Наборы критериев представляются:

— достаточно полными, поскольку значение обобщенного показателя $T^{(\alpha)}$ дает специалисту-макетировщику, производящему окончательный выбор компоновки из группы перспективных компоновок, удовлетворительное представление о степени достижения указанных целей (доход и эстетическая привлекательность издания) на этапе формирования структуры полосы издания;

— действенными, поскольку для макетировщика, оценивающего качество компоновки, понятен смысл критериев и соответствующих им показателей;

— разложимыми, что позволяет упростить процесс оценки качества компоновок, ограничиваясь оценкой их качества по какому-либо одному или по нескольким произвольно выбранным показателям;

— малоизбыточными (без существенного дублирования «учета различных аспектов последствий»);

— близкими к минимальным (оставляя размерность проблемы близкой к минимальной).

4. Выводы

Проведена унификация данных, характеризующих предметную область задачи формирования оригинал-макета издания. *Научная новизна* заключается в разработанных объектно-ориентированных моделях унифицированных структур данных для задачи формирования оригинал-макета издания. Эти модели могут быть использованы при проектировании базы данных издания.

Впервые введены два взаимосвязанных набора критериев, по которым оценивается качество компоновок статей на полосе издания и компоновок их фрагментов. Приведены процедуры вычисления значений показателей, соответствующих введенным критериям. Эти процедуры являются важной частью оптимизационных методов формирования структур полос издания.

Практическая значимость разработанных моделей и критериев определяется тем, что они являются неотъемлемой частью автоматизированной системы доредакционной подготовки изданий G1.

Список литературы: 1. Губницкая Ю. С., Гурьева Н. С. Методы управления потоками данных при обработке текстовой и графической информации // Системи обробки інформації. 2012. № 3. Т. 1. С. 127–133. 2. Губницкая Ю. С. Постановка задачи и алгоритм рационального размещения статей на полосе издания // Східно-Європейський журнал передових технологій. 2010. № 3. С. 12–19.

Поступила в редколлегию 23.11.2012

Губницкая Юлия Семеновна, аспирантка ХНУРЭ. Научные интересы: автоматизация доредакционной подготовки изданий. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 0503028627. E-mail: julgub@ukr.net

ІНТЕРАКТИВНА СИСТЕМА МОДЕЛЮВАННЯ ТА ОПТИМІЗАЦІЇ РЕЖИМІВ РОБОТИ КОМПРЕСОРНОГО ЦЕХУ З УРАХУВАННЯМ РОБОТИ СИСТЕМИ АВТОМАТИЧНОГО УПРАВЛІННЯ

Розглядається актуальна задача розробки методу побудови стохастичної області допустимих режимів роботи компресорного цеху з урахуванням обраної стратегії системи автоматичного управління компресорного цеху газотранспортної системи.

1. Вступ

На сьогодні накопичено значний досвід у галузі моделювання та оптимізації режимів транспорту й розподілу природного газу в газотранспортній системі (ГТС), тобто досвід розв'язування різноманітних класів задач проектування та планування режимів роботи ГТС [1]. Результати оптимізації структури і параметрів ГТС показали значні внутрішні резерви скорочення невиробничих затрат матеріальних та енергетичних ресурсів в ГТС.

Розроблені до нині методи оптимізації були детермінованими і не враховували неповноту та недостовірність апріорної інформації про структуру й параметри ГТС, реальні умови функціонування ГТС. Отримані з допомогою цих методів оптимальні розв'язки відповідали абсолютно точним значенням параметрів ГТС і абсолютно точним, конкретним значенням граничних умов, знаходячись при цьому, як правило, на границі допустимої області [2].

На практиці ж навіть незначні варіації параметрів чи граничних умов ГТС призводили не тільки до суттєвої зміни оптимального розв'язку, а й до виходу його за межі допустимої області. Природно, що такі "оптимальні" розв'язки виявились неприйнятними для оперативного-диспетчерського управління режимами роботи ГТС.

Метою даної роботи є дослідження математичних моделей та розробка методу побудови стохастичної області допустимих режимів роботи одного з ключових елементів ГТС – компресорного цеху (КЦ), а також моделювання режиму роботи КЦ залежно від стратегії, обраної системи автоматичного управління (САУ), що дозволяє підвищити ефективність прийняття рішень при плануванні та управлінні режимами роботи ГТС.

2. Математична модель КЦ

Розглянемо КЦ, який складається з K повнонапірних паралельно включених газоперекачувальних агрегатів (ГПА). ГПА – це основне силове устаткування ГТС. Він складається з нагнітача, у якому здійснюється стиск газу, і силового приводу, у якості якого на магістральних газопроводах (МГ) звичайно використовується газотурбінний двигун у складі газотурбінної установки (ГТУ). В даний час на компресорних станціях МГ звичайно застосовують відцентрові нагнітачі (ВН), для яких характерні велика продуктивність і невисокий ступінь стиску газу.

Інженерний розрахунок режиму роботи ГПА звичайно проводять на основі методики, викладеної в ОНТП-51-1-85 [3]. Математична модель фізичних процесів стиску газу в ВН і роботи ГТУ в даному випадку представляється системою нелінійних алгебраїчних рівнянь і нерівностей, що визначають обмеження на значення параметрів режиму роботи. Цей метод розрахунку є загальноприйнятим, але в існуючих програмних комплексах по розрахунку режимів роботи ГПА і КЦ часто використовуються або спрощені, або більш складні його модифікації, залежно від мети і ресурсів розрахунку.

Математична модель ВН. Основна витратно-напірна характеристика зв'язує сімох головних показників режиму роботи ВН: тиск газу на виході P_k (МПа) і вході P_n (МПа) ВН, їхнє відношення – ступінь стиску ε (-), об'ємну приведену продуктивність Q_{np} (м³/хв), частоту обертання валу n (про/хв), а також вхідну T_n (К) і вихідну T_k (К) температуру природного газу:

$$\varepsilon = P_k / P_n = \varepsilon(Q_{np}, (n/n_0)_{np}) = \left[1 + \left(\frac{n}{n_0} \right)^2 \left(\varepsilon_0^{\frac{m-1}{m}} - 1 \right) \right]^{\frac{m}{m-1}}, \quad (1)$$

де n_0 – номінальна частота обертання валу ВН; $\varepsilon_0 = \varepsilon(Q_{np}, 1)$ – паспортна характеристика ВН; m – показник політропи, який обчислюється досить складно, а значення температури газу на вході і виході нагнітача зв'язані співвідношенням

$$T_k = T_n \varepsilon^{\frac{m-1}{m}}. \quad (2)$$

Потужність $N_{цбн}$ ВН (кВт), що споживається ВН, знаходиться як

$$N_{цбн} = \frac{N_{iпр}(Q_{np}) \cdot \rho \cdot n_{np}^3}{0.95 \cdot \eta_m(Q_{np})},$$

тут $\eta_m(-)$ – механічний ККД, що наведено у паспортних характеристиках ВН; ρ (кг/м³) – густина природного газу за умовами на вході у нагнітач.

Наведена об'ємна Q_{np} і комерційна продуктивності q ВН зв'язані таким співвідношенням:

$$q = 60 \cdot 10^{-3} \frac{n}{n_0} Q_{np} \frac{\rho}{\rho_0}.$$

Для моделювання режиму ВН різних типів застосовується широко відома модель А.І.Степанова [4], заснована на сполученні аналітичних залежностей для політропних газодинамічних процесів і емпіричних характеристик, одержуваних для кожного ВН у процесі натурних іспитів і представлених у табличному або графічному вигляді. Ці характеристики будуються в наведеному вигляді для деякого фіксованого набору параметрів режиму (тиску, температури, газової постійної та ін).

Залежності $\varepsilon_0(Q_{np})$, $\eta_0(Q_{np})$ та $N_{iпр}(Q_{np})$ у програмних розрахункових комплексах звичайно представляють у виді багаточленів [2]:

$$\varepsilon_0(Q_{np}) = a_0 + a_1 Q_{np} + a_2 Q_{np}^2, \quad (3)$$

$$\eta_0(Q_{np}) = b_0 + b_1 Q_{np} + b_2 Q_{np}^2 + b_3 Q_{np}^3, \quad (4)$$

$$N_{iпр}(Q_{np}) = c_0 + c_1 Q_{np} + c_2 Q_{np}^2, \quad (5)$$

де коефіцієнти поліномів визначаються шляхом апроксимації паспортних характеристик, наданих заводом-виготовлювачем разом з конкретним екземпляром ВН або отриманих у результаті обробки даних експлуатації на компресорних станціях (КС).

На даний момент існує величезна кількість способів математичного моделювання режиму роботи ВН. Розглянемо декілька з них.

Залежності $\varepsilon_0(Q_{np})$, $\eta_0(Q_{np})$ та $N_{iпр}(Q_{np})$ можуть бути представлені поліномами не 2-3 ступеня, як показано у формулах (3)-(5), але 4-ї, 5-ї та ще вищих ступенів, що збільшує точність відповідності модельних значень паспорту ВН.

Розвитком цього напрямку є апроксимація паспортних залежностей сплайнами, що обумовлює ще більшу відповідність моделі паспорту ВН.

З формальної точки зору залежність (1) описує деяку поверхню, що може бути представлена двовимірним поліномом:

$$\varepsilon = \sum_{i=1}^a Q_{np}^i \sum_{j=1}^b d_{ij} n_{np}^j,$$

де коефіцієнти d_{ij} отримуються шляхом відповідної апроксимації паспортних даних.

Використання залежності (1) пов'язане із значними витратами машинного часу. Тому обчислення ε може виконуватися з залученням так званих універсальних функцій

$$n_{np}^2 \varepsilon_0(Q_{np}) = \omega(\varepsilon),$$

де $\omega(\varepsilon)$ апроксимується, наприклад поліномом 3-го ступеня. В цьому випадку співвідношення (2) також може бути замінено на простіше:

$$T_k = T_n \left[1 + \frac{k-1}{k \cdot \eta(Q_{пр})} \cdot \omega(\varepsilon) \right].$$

Альтернативний підхід до розрахунку режиму роботи ВН полягає у використанні так званих безвимірних величин, як це робиться у СМНПО ім. Фрунзе (м. Суми), що є головним виробником та постачальником ВН в Україні.

Математична модель ГТУ. Основою для розрахунків режимів роботи ГТУ є тепло-технічні характеристики, визначені в ході приймально-здаючих іспитів силової турбіни заводом-виготовлювачем і зафіксовані у формулярі кожного конкретного екземпляра ГТУ. Вони являють собою таблиці значень основних контрольованих параметрів роботи силової турбіни на 100%, 75%, 50% і 25% номінальної потужності при деяких фіксованих умовах (температурі повітря на вході двигуна, атмосферному тиску), названих умовами приведення. Методика роботи з цими характеристиками, представленими в таблично-графічному виді, наводиться в довідковій літературі (наприклад [5]) або дається безпосередньо заводом-виготовлювачем. Для комп'ютерного моделювання й автоматизованих розрахунків вихідні дані апроксимуються, у результаті цього математичну модель ГТУ можна представити у виді системи нелінійних рівнянь. Розглянемо основне співвідношення моделі ГТУ.

Коефіцієнти приведення параметрів ГТУ:

$$\alpha = \sqrt{\frac{t_{пр} + 273}{t_{кнт} + 273}}, \quad \beta = \frac{p_{a0}}{p_{атм}},$$

де $t_{кнт}$ і $t_{пр}$ (К) – фактичне та паспортне значення температури повітря на вході в

компресор ГТУ; $p_{атм}$ і p_{a0} (мм рт.ст.) – фактичне і паспортне значення атмосферного тиску.

Наведені значення потужності N (кВт) і масової витрати паливного газу G (кг/рік) зв'язані з фактичними значеннями співвідношеннями:

$$N_{ГТУ пр} = N_{ГТУ} \cdot \alpha \cdot \beta, \quad G = G_{ном} \cdot (1 - 0.75(1 - N_{ГТУ пр})) \cdot \alpha \cdot \beta.$$

Залежність між $N_{ГТУ пр}$ і $G_{пр}$ визначається довідковим співвідношенням [1]:

$$G_{пр} = 1 - 0.75(1 - N_{ГТУ пр})$$

або за результатами апроксимації паспортних теплотехнічних характеристик конкретного екземпляра ГТУ як $G_{пр} = e_0 + e_1 N_{ГТУ пр} + e_2 N_{ГТУ пр}^2$.

Режим роботи ГТУ обмежений максимальною частотою обертання компресора високого тиску, що залежить від температури повітря на вході в компресор двигуна t_a . Апроксимація залежності розташовуваної потужності двигуна $N_{max}(t_a)$ дає таке обмеження на значення потужності ГТУ N :

$$N \leq N_{max}(t_a) = \begin{cases} 1.2N_{ном}, & t_a + \delta t \leq -5^\circ \text{C}, \\ h_0 - h_1 t_a, & t_a + \delta t > -5^\circ \text{C}, \end{cases} \quad (6)$$

де коефіцієнти h_0 і h_1 визначаються для кожного типу ГТУ, а $\delta t \approx 5^\circ \text{C}$ – виправлення на різницю температури повітря атмосферного і на вході ГТУ.

Для математичного опису зв'язку між ГТУ і ВН як частин одного ГПА використовують принцип рівності й одержуваної потужності [3]:

$$N = N_{ВН} + N_{мех},$$

де $N_{мех}$ – механічні втрати на муфті приводу, обумовлені залежно від типу ВН.

3. Побудова області допустимих режимів функціонування КЦ

При постановці та розв'язанні задач ОДР функціонування ГПА використовуються два види інформації: оперативна та нормативно-довідкова [5]. Оперативна інформація включає в себе результати прямих та опосередкованих вимірів параметрів газових потоків і параметрів, що характеризують режими роботи ГПА.

В свою чергу, всі прямі виміри поділяються на виміри неперервних параметрів (тиску, витрат, температури, вуглеводневого складу природного газу і т.п.) та дискретних параметрів (стану запірних кранів “відкрито-закрито” та ін.).

Всі опосередковані виміри відносяться до вимірів неперервних величин і являють собою обчислювані значення детермінованих функцій, аргументами яких є результати прямих вимірів.

Відомо [6], що в будь-яких системах вимірювань виникають похибки, пов’язані як з похибками первинних датчиків, так і з перешкодами в каналах зв’язку. Ці похибки (помилки) мають випадковий характер з відомим законом розподілу (як правило, нормальним) та відомими параметрами – математичним очікуванням та дисперсією.

За відсутності систематичної похибки математичне очікування помилок приймається рівним нулеві, а дисперсія помилок вимірювання визначається класом точності приладу та його діапазоном вимірювань. Таким чином, вся оперативна інформація, що отримується у результаті прямих чи опосередкованих вимірювань, за своєю природою є стохастичною.

Нормативно-довідкова інформація (НДІ), яка необхідна для розв’язування задачі побудови ОДР функціонування ГПА, включає в себе інформацію про параметри математичних моделей ГПА. Основними параметрами моделей ГПА є параметри термодинамічних характеристик ВН та ГТУ.

Параметри математичних моделей термодинамічних характеристик ВН та ГТУ оцінюються або за паспортними характеристиками, або за експериментальними даними. Паспортні характеристики справедливі для всієї партії ВН чи ГТУ даної серії та даного типу, але можуть суттєво відрізнятися для кожного конкретного екземпляра.

Отримані оцінки параметрів термодинамічних характеристик ВН та ГТУ за експериментальними даними за своєю суттю є випадковими величинами, оскільки оцінюються за результатами скінченної кількості прямих та опосередкованих вимірів повітряних та газових потоків, що мають природні похибки вимірювань.

Об’єктивне урахування стохастичної природи оперативної інформації є необхідною умовою при оцінці фактичних параметрів моделей ГПА, а також при оцінці його технічного стану.

Таким чином, при розв’язуванні задач оперативного контролю та керування режимами роботи ГПА, встановлених на КС магістральних газопроводів, виникає проблема контролю відповідності знаходження фактичної робочої точки (РТ) ВН ГПА в ОДР, оскільки відомо [5], що стійкість, надійність та безпека режимів роботи ВН ГПА залежать від ступеня віддаленості РТ від границь ОДР, а вихід РТ за межі ОДР розглядається як аварійна ситуація.

3.1. Побудова ОДР ГПА

ОДР роботи ВН ГПА визначається такою системою нерівностей:

– границя помпажної зони ВЦН та гранично допустима об’ємна продуктивність

$$Q_{\min} \leq Q_{\text{np}} \leq Q_{\max}; \quad (7)$$

– мінімальна та максимальна кількість обертів приводу ВН

$$n_{\min} \leq n \leq n_{\max}; \quad (8)$$

– максимальна (наявна) потужність, яка обчислюється згідно з формулою (6);

– максимальний вихідний тиск, що визначається міцністю труб:

$$P_{\text{вих}} \leq P_{\max}; \quad (9)$$

– максимальна температура газу на виході ВЦН, що визначається властивостями ізоляційного покриття

$$T_{\text{вих}} < T_{\max}. \quad (10)$$

Вхідними даними для побудови ОДР і обчислення РТ для кожного фіксованого моменту часу є виміри значення параметрів газового потоку на вході та виході ВН, а також виміряні оберти приводу:

$$\tilde{T}_{\text{вх}} = T_{\text{вх}} + \Delta T_{\text{вх}}, \tilde{T}_{\text{вих}} = T_{\text{вих}} + \Delta T_{\text{вих}}, \tilde{P}_{\text{вх}} = P_{\text{вх}} + \Delta P_{\text{вх}}, \tilde{P}_{\text{вих}} = P_{\text{вих}} + \Delta P_{\text{вих}}, \tilde{n} = n + \Delta n,$$

де $T_{\text{вх}}, P_{\text{вх}}, T_{\text{вих}}, P_{\text{вих}}, n$ – істинні (невідомі) значення параметрів газового потоку та кількості обертів нагнітача, а $\Delta T_{\text{вх}}, \Delta T_{\text{вих}}, \Delta P_{\text{вх}}, \Delta P_{\text{вих}}$ – сумарні випадкові помилки вимірів відповідних

величин і помилки в каналах зв'язку, статистичні властивості яких визначаються в ході метрологічних дослідів.

Паспортні характеристики ВН – коефіцієнти апроксимації визначаються у відповідності з (3)-(5).

Гранично допустима ймовірність виходу РТ за межі ОДР ($a \ll 1$).

Формалізація обмежень (6)–(10) в явному вигляді, тобто у вигляді системи нерівностей викликає складнощі при чисельному розв'язанні задач розрахунку чи оптимізації роботи ГПА. Тому більш ефективним є підхід, що полягає у зведенні цієї системи до однієї нерівності з використанням принципів побудови ОДР ГПА (рис. 1). Якщо границі 1-4 жорстко пов'язані між собою (тобто при зміні параметрів (P_n, T_n, t_a) зберігають відносне положення), то положення границі 5, яке визначається значенням t_a , не залежить від положення границь 1-4 (див. рис. 1).

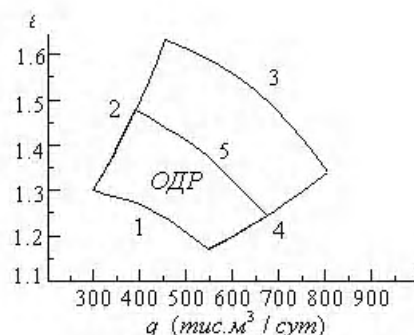


Рис. 1. Загальний вигляд ОДР ГПА

На рис.1 границі 1 та 3 відповідають обмеженням $n_{пр \min}$, $n_{пр \max}$; границі 2 та 4 - $Q_{пр \min}$, $Q_{пр \max}$, а границя 5 – обмеженню на потужність ГТУ $N_{\max}$.

При розв'язуванні більшості задач оперативно-диспетчерського управління достатньо вміти розраховувати ліву та праву границі ОДР q^- і q^+ , що відповідають P_k при відомих P_n , T_n і t_a .

Знайшовши пари $(Q_{пр}^-, n_{пр}^-)$ і $(Q_{пр}^+, n_{пр}^+)$, що відповідають точкам перетину горизонтальної лінії $\varepsilon = P_k / P_n = \text{const}$ з границями ОДР, можна відновлювати граничні значення q^- та q^+ :

$$q = 60 \cdot 10^{-3} n_{пр} \sqrt{\frac{Z_n T_n R}{Z_{пр} T_{пр} R_{пр}}} Q_{пр} \frac{10^6 P_n}{\rho_0 Z_n R T_n} = \frac{60 \cdot 10^3 n_{пр} Q_{пр} P_n}{\rho_0 \sqrt{Z_n R T_n} \sqrt{Z_{пр} T_{пр} R_{пр}}},$$

звідси

$$\begin{aligned} q^-(P_n, T_n, P_k) &= n_{пр}^- Q_{пр}^- \frac{60 \cdot 10^3 P_n}{\rho_0 \sqrt{Z_n R T_n} \sqrt{Z_{пр} T_{пр} R_{пр}}}, \\ q^+(P_n, T_n, P_k) &= n_{пр}^+ Q_{пр}^+ \frac{60 \cdot 10^3 P_n}{\rho_0 \sqrt{Z_n R T_n} \sqrt{Z_{пр} T_{пр} R_{пр}}}. \end{aligned} \quad (11)$$

Зв'язок потужності, яка віддається ГТУ, з характерними параметрами режиму роботи ВЦН $Q_{пр}$ та $n_{пр}$ визначається співвідношенням:

$$N = \frac{10^3 n_{пр}^3 P_n}{0.95 \eta_M Z_n R T_n} \left(\frac{Z_{пр} R_{пр} T_{пр}}{Z_n R T_n} \right)^{3/2} v(Q_{пр}) + N_{\text{мех}}. \quad (12)$$

Розв'язок системи (1), (12) при $N = N_{\max}(t_a)$ відносно змінних $Q_{пр}$ та $n_{пр}$ дає шукану пару (Q_N^+, n_N^+) .

Змінюючи ε від $\frac{P_{k,\min}}{P_n}$ до $\frac{P_{k,\max}}{P_n}$ з достатньо малим кроком, отримаємо множину точок, що репрезентуватимуть ОДР ГПА.

3.2. Побудова стохастичної ОДР ГПА

Якщо значення параметрів газу на вході та виході ВН відомі не точно, то і всі знайдені за ними граничні значення ОДР будуть досить наближеними. Це звичайне явище при оперативно-диспетчерському контролі стану ГПА, оскільки значення P_n , T_n , і t_a змінюються з деякою природною похибкою. Відповідно до [7] вважатимемо закони розподілу P_n , T_n , і t_a нормальними з параметрами (m, σ) .

Вважаючи випадкові величини P_n , T_n , і t_a незалежними, маємо:

$$m_q = q = q(m_{P_n}, m_{T_n}, m_{t_a}),$$

$$\sigma_q^2 = \left(\frac{\partial q}{\partial P_n} \right)_{m_q}^2 \sigma_{P_n}^2 + \left(\frac{\partial q}{\partial T_n} \right)_{m_q}^2 \sigma_{T_n}^2 + \left(\frac{\partial q}{\partial t_a} \right)_{m_q}^2 \sigma_{t_a}^2. \quad (13)$$

В зв'язку зі значною складністю вирахування аналітичних залежностей для похідних нелінійних неявних рівнянь математичної моделі ГПА значення похідних у виразі (13) будемо вираховувати чисельно [7].

Для графічного представлення ОДР необхідно побудувати оцінки границь довірчих областей, в яких із заданою ймовірністю P знаходяться істинні межі ОДР. При $P=0.998$ довірчі інтервали представляються як $[q^- - 3\sigma_{q^-}, q^- + 3\sigma_{q^-}]$ і $[q^+ - 3\sigma_{q^+}, q^+ + 3\sigma_{q^+}]$, де σ_{q^-} і σ_{q^+} знаходяться у відповідності з (13). З'єднуємо кінці цих інтервалів для різних P_k із діапазону можливих значень та отримуємо границі довірчих інтервалів, які зображено на рис. 2 пунктирними лініями. На рис. 3 показано результати

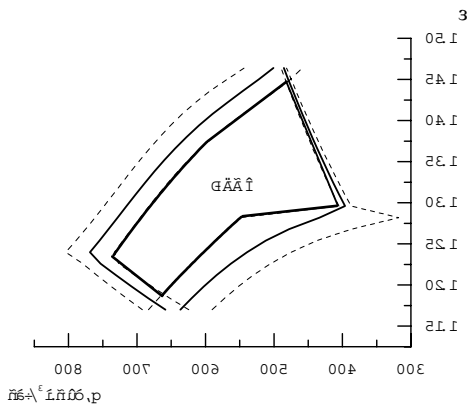


Рис. 2. Зміна розміру та геометрії ОДР ГПА при похибці виміру параметрів від точності вимірів параметрів режиму його роботи у межах 2%

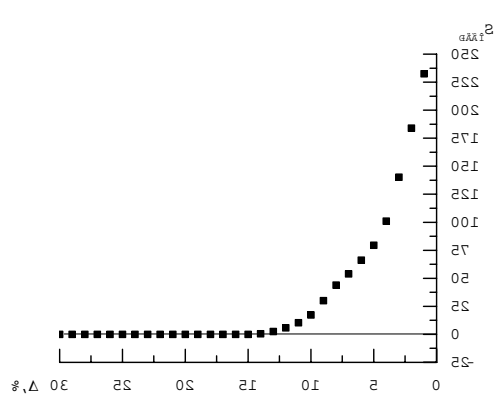


Рис. 3. Залежність площі ОДР ГПА

3.3. Математична модель компресорного цеху

Компресорна станція зазвичай складається з декількох компресорних цехів, що представляють собою завершені лінії компримування газу. При розв'язуванні задач оптимізації режимів роботи ГТС практично немає необхідності в детальних відомостях про поточкорозподіл за всіма технологічними елементами КЦ, тобто не обов'язково знати значення параметрів газових потоків у будь-якій точці технологічної схеми цеху. Цілком достатньо вміти пов'язувати значення тисків, температур та витрат газу на вході та виході цеху з параметрами роботи основного та допоміжного обладнання. Тому для розв'язування таких

задач пропонується використовувати моделі КЦ і КС, що не вимагають значних обчислювальних ресурсів, але дозволяють отримувати достатньо адекватні моделі. Для побудови таких моделей пропонуємо використати стандартний прийом, що застосовується для спрощення гідравлічного розрахунку складних газопроводів – метод еквівалентності [9]. Схема отриманої моделі КП, оснащеного N паралельно увімкненими ГПА, зображена на рис.4.

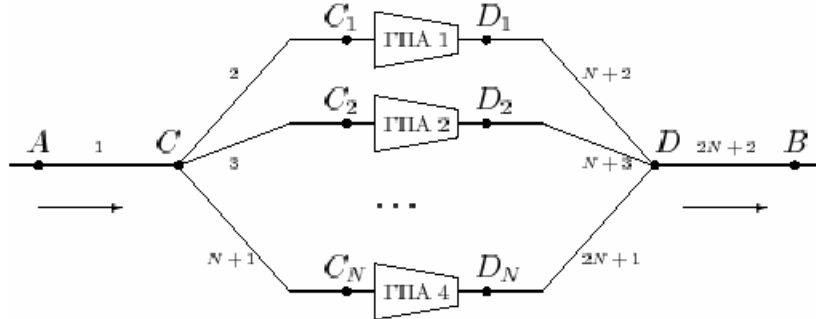


Рис. 4. Схема КЦ, оснащеного N паралельно увімкненими ГПА

Тут точки А та В відповідають місцям встановлення датчиків тиску та температури газу на вході й виході КЦ, точки С та D – на входньому та вихідному колекторах цеху, а $C_1, \dots, C_N$ та $D_1, \dots, D_N$ – на вході та виході кожного ГПА. Ділянки АС, $CC_1, \dots, CC_N$, $DD_1, \dots, DD_N$ і DB, які відповідають реальним частинам трубопровідної об'язки, що зв'язує ГПА з АПО пристроями підготовки паливного та імпульсного газу та ін., замінені на еквівалентні з довільно обраними діаметрами D_{ji} та довжинами L_{ji} ($i = 1, \dots, 2N+2$), але з гідравлічним опором λ_{ji} , оціненим попередньо за замірами фактичних параметрів режиму роботи КЦ.

Отже, при моделюванні стаціонарного режиму роботи КЦ моделі ГПА пропонуємо об'єднати з допомогою моделей еквівалентних ділянок трубопроводів згідно з прийнятою схемою (див. рис 4):

$$P_{ki}^2 - P_{ni}^2 = c\Delta\lambda_{ji}Z_{cpi}T_{cpi}L_{ji}D_{ji}^{-5}q_i^2 = \Lambda_{ji}q_i^2, \quad i = 1, \dots, 2N+2, \quad (14)$$

$$T_{ki} = T_{ni} \exp(-a_{ji}L_{ji}/q_i), \quad i = 1, \dots, N+1. \quad (15)$$

Оскільки на ділянці АВ нема відбору газу, а витрати у підвідному та відвідному ВН ГПА рівні, можна записати:

$$q_2 = q_{N+2}; \quad \dots \quad q_{N+1} = q_{2N+1}, \quad (16)$$

$$q_{КЦ} = q_1 = q_{2N+2} = \sum_{k=1}^N q_{1+k} = \sum_{k=1}^N q_{N+1+k}, \quad (17)$$

а умова рівності тисків у вузлах (у припущенні про статичність потокорозподілу) дає:

$$P_{k1} = P_{n2} = \dots = P_{nN+1}; \quad (18)$$

$$P_{n2N+2} = P_{kN+2} = \dots = P_{k2N+1}. \quad (19)$$

Точність отриманих оцінок параметрів режиму цілком достатня для використання цієї моделі КЦ при розв'язуванні оптимізаційних задач оперативно-диспетчерського управління ГТС [9].

3.4. Побудова ОДР компресорного цеху

Обмеження на режими роботи КЦ природньо визначаються технологічними обмеженнями на режими роботи ГПА, установки охолодження газу (УохГ), установки очищення газу (УочГ), АПО та ін. При побудові ОДР КЦ обмеження на режимні параметри УохГ, УочГ, АПО можна не враховувати, оскільки допоміжне технологічне обладнання зазвичай проектується з таким розрахунком, щоб забезпечувати нормальне функціонування КЦ при всіх можливих режимах роботи силового обладнання (ГПА). Як і ОДР ГПА, ОДР КЦ може використовуватись як для візуального контролю фактичного режиму роботи цеху, так і для аналітичного аналізу запасу ресурсів КЦ “по керуванню” на етапі планування. В літературі

згадки про використання ОДР КЦ в системах оперативно-диспетчерського управління зустрічаються досить рідко, на відміну від ОДР ГПА. Це пов'язано, скоріш за все, з обчислювальними складностями, які виникають при її побудові. Окрім того, сам метод її побудови ніде не формалізовано. Той факт, що згідно з моделлю КЦ (14)-(19) продуктивність цеху дорівнює сумі витрат через нагнітачі всіх працюючих ГПА КЦ, дозволяє розробити наступний підхід до побудови ОДР КЦ. Не втрачаючи загальності, для початку розглянемо задачу знаходження правої границі ОДР КЦ, що відповідає максимальній продуктивності цеху $q_{\text{КЦ}}^+$; а ліву границю, що відповідає мінімальній продуктивності $q_{\text{КЦ}}^-$ будемо шукати за аналогією. Максимальна продуктивність $q_{\text{КЦ}}^+$, яку зможе забезпечити КЦ при заданих значеннях температури газу на його вході та виході (P_A, T_A, P_B, T_B), очевидно, буде сумою максимально допустимих витрат через кожен ГПА $q_{\text{КЦ}}^+$. Проблема знаходження $q_{\text{КЦ}}^+$ полягає в тому, що витрати ГПА $q_{\text{КЦ}}^+$ залежать від тисків на вході і виході ГПА та температури газу на його вході. Тому виникає необхідність розв'язувати задачу розрахунку усталеного поточкорозподілу в КЦ, яка зводиться до розв'язування системи рівнянь, записаної згідно з моделлю КЦ (14)-(19), а також з урахуванням (11):

$$P_A^2 - P_C^2 = \lambda_{AC} (q_{\text{КЦ}}^+)^2; \quad (20)$$

$$P_C^2 - P_{C_i}^2 = \lambda_{CC_i} (q_i^+)^2; \quad i = 1, \dots, N; \quad (21)$$

$$P_{D_i}^2 - P_D^2 = \lambda_{D_iD} (q_i^+)^2; \quad i = 1, \dots, N; \quad (22)$$

$$P_D^2 - P_B^2 = \lambda_{DB} (q_{\text{КЦ}}^+)^2; \quad i = 1, \dots, N; \quad (23)$$

$$T_C = T_A \exp(-\alpha_{AC} (q_{\text{КЦ}}^+)^{-1}); \quad (24)$$

$$T_{C_i} = T_C \exp(-\alpha_{CC_i} (q_i^+)^{-1}); \quad i = 1, \dots, N; \quad (25)$$

$$q_i^+ = f_i^+(P_{C_i}, T_{C_i}, P_{D_i}); \quad i = 1, \dots, N; \quad (26)$$

$$q_{\text{КЦ}}^+ = \sum_{i=1}^N q_i^+. \quad (27)$$

Тут процедура обчислення f_i^+ задана в п. 3.1, а значення P_A, T_A, P_B , вважаються відомими. Система (20)-(27) має $(4N+4)$ рівнянь і $(4N+3)$ незалежних змінних, тобто одне з рівнянь є "зайвим", наприклад (27), оскільки значення $q_{\text{КЦ}}^+$ не є незалежною змінною, а повністю визначається значеннями q_i^+ . Простою підстановкою можна легко звести систему (20)-(27) до системи, що складається лише з N рівнянь з N невідомими. На першому кроці отримаємо:

$$P_A^2 - P_{C_i}^2 = \lambda_{AC} (q_{\text{КЦ}}^+)^2 + \lambda_{CC_i} (q_i^+)^2, \quad i = 1, \dots, N; \quad (28)$$

$$P_{D_i}^2 - P_B^2 = \lambda_{D_iD} (q_i^+)^2 + \lambda_{DB} (q_{\text{КЦ}}^+)^2, \quad i = 1, \dots, N; \quad (29)$$

$$T_{C_i} = T_A \exp(-\alpha_{AC} (q_{\text{КЦ}}^+)^{-1}) \exp(-\alpha_{CC_i} (q_i^+)^{-1}), \quad i = 1, \dots, N; \quad (30)$$

$$q_i^+ = f_i^+(P_{C_i}, T_{C_i}, P_{D_i}), \quad i = 1, \dots, N; \quad (31)$$

$$q_{\text{КЦ}}^+ = \sum_{i=1}^N q_i^+, \quad (32)$$

а потім виразимо $P_{C_i}, T_{C_i}, P_{D_i}$ із (28)-(30) і підставимо їх і $q_{\text{КЦ}}^+$ з (32) у вираз для q_i^+ (31). Отримаємо:

$$q_i^+ = f_i^+(\phi_1(q^+), \phi_2(q^+), \phi_3(q^+)); \quad i = 1, \dots, N; \quad (33)$$

де $q^+ = (q_1^+, \dots, q_N^+)$, а ϕ_1, ϕ_2 і ϕ_3 – деякі функції, одержані в результаті підстановок. Іншими словами, отримуємо систему вигляду

$$q^+ = F_{\text{КЦ}}^+(q^+). \quad (34)$$

Розв'язування (34) відносно вектора q^+ розмірності N можливе багатьма способами, проте сам вигляд рівнянь дає можливість використовувати метод простих ітерацій [7], обчислювальну схему якого можна представити у вигляді

$$\mathbf{q}_{k+1}^+ = \mathbf{F}_{\text{КЦ}}^+(\mathbf{q}_k^+), \quad (35)$$

тут k – номер кроку ітерації, а як $\mathbf{q}_0^+$ можна взяти вектор значень номінальної продуктивності кожного ГПА КЦ, а обчислення продовжувати до тих пір, доки не виконуватиметься нерівність $\|\mathbf{q}_{k+1}^+ - \mathbf{q}_k^+\| < \xi$, де ξ – деяка достатньо мала величина. Ітераційна схема (35) буде збіжною, лише якщо $\mathbf{F}$ – оператор, що стискає. Аналітичне доведення цього факту є досить важким, зважаючи на складність функцій f_i^+ , ϕ_1 , ϕ_2 і ϕ_3 . Проте опосередкованим доведенням можна вважати той факт, що в ході чисельного розв’язування системи для різних значення P_A , T_A , P_B та їх кількості N метод завжди був збіжний і давав розумні, з фізичної точки зору, результати.

Отже, в результаті розв’язування системи (34) будуть знайдені значення $\mathbf{q}_i^+$, за якими згідно з (32) легко знаходиться шукане $\mathbf{q}_{\text{КЦ}}^+$. Аналогічно можна знайти і $\mathbf{q}_{\text{КЦ}}^-$.

Результатом проведеного наукового дослідження стала програма WCompressor, написана мовою програмування C# на платформі .NET 3.0, яка здійснює побудову стохастичної області допустимих режимів роботи компресорного цеху з ймовірністю 0,9985 (рис. 5).

Передбачена можливість змінювати кількість зайнятих ГПА та параметри мережі, що їх з’єднує: гідравлічний опір та коефіцієнти пониження температури на ділянках.

Цінним модулем програми є можливість моделювати роботу компресорного цеху (рис. 6, а, б) в умовах випадкової зміни параметрів навколишнього середовища та газу, що перекачується, залежно від обраної стратегії системи автоматичного управління. При цьому можна чітко відслідковувати рух «робочої точки» в межах ОДР, а також режим роботи ГПА, а саме: його ККД, потужність та витрату палива.

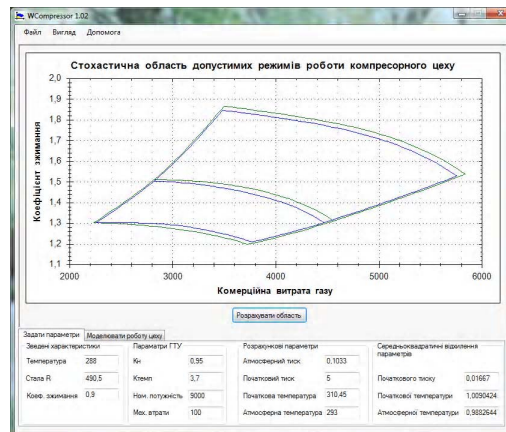


Рис. 5. Головне вікно програми

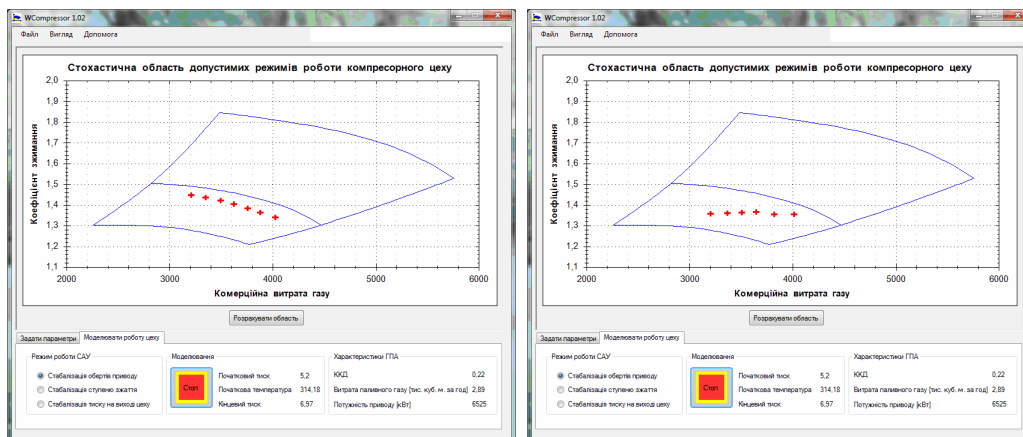


Рис. 6. Моделювання режимів роботи КЦ при стратегії «Стабілізація обертів приводу ВН» (а) та «Стабілізація ступеня зжаття» (б)

Основні стратегії режимів роботи САУ, які застосовуються на КС МГ, є «Стабілізація обертів приводу ВН», «Стабілізація ступеня зжаття» та «Стабілізація тиску на виході з КЦ». Зазначимо, що стратегія управління по кожному виду ГПА відрізняється та залежить від багатьох факторів (технічних, економічних).

4. Висновки

Розглянуто актуальну задачу розробки методу побудови стохастичної області допустимих режимів роботи компресорного цеху з урахуванням обраної стратегії системи автоматичного управління, який було реалізовано у вигляді програми WCompressor. Наведено, що стохастичний підхід дозволяє більш адекватно описувати реальні режими роботи технологічно обладнання ГТС, більш точно (візуально та програмно) контролювати границі ОДР. Запропонований метод може бути ефективно застосований для моделювання та оптимізації режимів роботи ГПА окремо та КЦ у цілому, а також для прогнозування та своєчасного застереження виходу РТ із ОДР.

Список літератури: 1. Розгонюк В.В, Хачикян Л.А. Эксплуатационники газонафтового комплекса. Довідник. К.: Росток, 1998. 429с. 2. Евдокимов А.Г. Минимизация функций. Х.: Вища шк., 1977. 288с. 3. Магистральные трубопроводы. Часть 1. Газопроводы : ОНТП-51-1-84. [Действительный от 1986-01-01]. К. : Госстандарт Украины, 1999. 95 с. (Отраслевые нормы технологического проектирования). 4. Трубопроводный транспорт газа / [Ковалко М. П., Грудз В. Я., Михалків В. Б. та ін.]. К. : Агентство з раціонального використання енергії та екології, 2002. 600 с. 5. Константинова И. М. Математическое моделирование технологических объектов магистрального транспорта газа / Константинова И. М. М. : Недра, 1988. 192 с. 6. Орловский С.А. Проблемы принятия решений при нечеткой исходной информации. М.: Наука, 1981. 206 с. 7. Измаилов А. Ф. Численные методы оптимизации: учеб. пособие / А. Ф. Измаилов, М. В. Солодов. М. : Физматлит, 2005. 304 с. 8. Тевяшев А. Д. Стохастические модели и методы оперативно-диспетчерского управления газотранспортными системами / А. Д. Тевяшев // АСУ и приборы автоматики. 2004. № 131. С. 22–32. 9. Трубопроводные системы энергетики. Управление развитием и функционированием / [Тевяшев А. Д., Ставорский Е. Р., Сухарев М. Г. и др.]; под ред. А.Д. Тевяшева. Новосибирск : Наука, 2002. 586 с.

Поступила в редколлегию 14.11.2012

Ісвєлєв Світлана Миколєївна, канд. техн. наук, доцент каф. прикладної математики ХНУРЕ. Наукові інтереси: математичне моделювання складних систем. Хобі: нумізматика та активний відпочинок. Адреса: Україна, 61166, Харків, пр. Леніна, 14, тел. 050-605-74-46.

УДК 007.5; 004.85

С.Ф. ЧАЛЫЙ, Е.О. БОГАТОВ, Д.Г. МЕЛЕШКО

ТЕХНОЛОГИЯ ПРЕДВАРИТЕЛЬНОЙ СТРУКТУРИЗАЦИИ ЖУРНАЛОВ РЕГИСТРАЦИИ СОБЫТИЙ СЛАБОСТРУКТУРИРОВАННЫХ БИЗНЕС-ПРОЦЕССОВ

Предлагается технология предварительной обработки журналов регистрации событий для задач интеллектуального анализа бизнес-процессов. Журналы регистрации событий фиксируют информацию о последовательности событий, которые происходят при функционировании бизнес-процессов. Предлагаемая технология позволяет расширить область применения методов process mining для слабоструктурированных бизнес-процессов.

1. Актуальность

В настоящее время активно развиваются методы интеллектуального анализа бизнес-процессов (БП), направленные на построение моделей таких процессов на основе анализа их журналов регистрации событий (ЖРС). Последние обычно формируются соответствующей информационной системой и фиксируют события, отражающие последовательность действий выполняющихся бизнес-процессов. К задачам данного научного направления относят разработку моделей бизнес-процессов, выявление узких мест в таких процессах, формирование набора бизнес-правил, отражающих зависимости между процедурами процесса и ограничениями на его выполнение.

Основні стратегії режимів роботи САУ, які застосовуються на КС МГ, є «Стабілізація обертів приводу ВН», «Стабілізація ступеня зжаття» та «Стабілізація тиску на виході з КЦ». Зазначимо, що стратегія управління по кожному виду ГПА відрізняється та залежить від багатьох факторів (технічних, економічних).

4. Висновки

Розглянуто актуальну задачу розробки методу побудови стохастичної області допустимих режимів роботи компресорного цеху з урахуванням обраної стратегії системи автоматичного управління, який було реалізовано у вигляді програми WCompressor. Наведено, що стохастичний підхід дозволяє більш адекватно описувати реальні режими роботи технологічно обладнання ГТС, більш точно (візуально та програмно) контролювати границі ОДР. Запропонований метод може бути ефективно застосований для моделювання та оптимізації режимів роботи ГПА окремо та КЦ у цілому, а також для прогнозування та своєчасного застереження виходу РТ із ОДР.

Список літератури: 1. Розгонюк В.В, Хачикян Л.А. Эксплуатационники газонафтового комплекса. Довідник. К.: Росток, 1998. 429с. 2. Евдокимов А.Г. Минимизация функций. Х.: Вища шк., 1977. 288с. 3. Магистральные трубопроводы. Часть 1. Газопроводы : ОНТП-51-1-84. [Действительный от 1986-01-01]. К. : Госстандарт Украины, 1999. 95 с. (Отраслевые нормы технологического проектирования). 4. Трубопроводный транспорт газа / [Ковалко М. П., Грудз В. Я., Михалків В. Б. та ін.]. К. : Агентство з раціонального використання енергії та екології, 2002. 600 с. 5. Константинова И. М. Математическое моделирование технологических объектов магистрального транспорта газа / Константинова И. М. М. : Недра, 1988. 192 с. 6. Орловский С.А. Проблемы принятия решений при нечеткой исходной информации. М.: Наука, 1981. 206 с. 7. Измаилов А. Ф. Численные методы оптимизации: учеб. пособие / А. Ф. Измаилов, М. В. Солодов. М. : Физматлит, 2005. 304 с. 8. Тевяшев А. Д. Стохастические модели и методы оперативно-диспетчерского управления газотранспортными системами / А. Д. Тевяшев // АСУ и приборы автоматики. 2004. № 131. С. 22–32. 9. Трубопроводные системы энергетики. Управление развитием и функционированием / [Тевяшев А. Д., Ставорский Е. Р., Сухарев М. Г. и др.]; под ред. А.Д. Тевяшева. Новосибирск : Наука, 2002. 586 с.

Поступила в редколлегию 14.11.2012

Ісвєлєв Світлана Миколєївна, канд. техн. наук, доцент каф. прикладної математики ХНУРЕ. Наукові інтереси: математичне моделювання складних систем. Хобі: нумізматика та активний відпочинок. Адреса: Україна, 61166, Харків, пр. Леніна, 14, тел. 050-605-74-46.

УДК 007.5; 004.85

С.Ф. ЧАЛЫЙ, Е.О. БОГАТОВ, Д.Г. МЕЛЕШКО

ТЕХНОЛОГИЯ ПРЕДВАРИТЕЛЬНОЙ СТРУКТУРИЗАЦИИ ЖУРНАЛОВ РЕГИСТРАЦИИ СОБЫТИЙ СЛАБОСТРУКТУРИРОВАННЫХ БИЗНЕС-ПРОЦЕССОВ

Предлагается технология предварительной обработки журналов регистрации событий для задач интеллектуального анализа бизнес-процессов. Журналы регистрации событий фиксируют информацию о последовательности событий, которые происходят при функционировании бизнес-процессов. Предлагаемая технология позволяет расширить область применения методов process mining для слабоструктурированных бизнес-процессов.

1. Актуальность

В настоящее время активно развиваются методы интеллектуального анализа бизнес-процессов (БП), направленные на построение моделей таких процессов на основе анализа их журналов регистрации событий (ЖРС). Последние обычно формируются соответствующей информационной системой и фиксируют события, отражающие последовательность действий выполняющихся бизнес-процессов. К задачам данного научного направления относят разработку моделей бизнес-процессов, выявление узких мест в таких процессах, формирование набора бизнес-правил, отражающих зависимости между процедурами процесса и ограничениями на его выполнение.

Основное ограничение для существующих методов интеллектуального анализа процессов состоит в том, что исходные данные для построения модели должны быть логически структурированы таким образом, чтобы отображать «алгоритмы» выполнения отдельных частей анализируемого процесса в виде последовательности событий. Формирование указанных структурированных журнальных файлов осуществляется только в ВРМ - системах в автоматическом режиме.

В то же время, существует значительное количество организаций, в которых бизнес-процессы слабо структурированы. При этом обычно используются не процессно-ориентированные корпоративные информационные системы, что вызывает трудности в поддержке, усовершенствовании, реинжиниринге таких процессов. Текущие действия в данных информационных системах протоколируются с метками времени вне зависимости от того, к какому процессу принадлежит фиксируемое действие. Иными словами, регистрационные файлы в таких системах содержат информацию, необходимую для построения процессной модели, отражающей реально выполняющиеся бизнес-процессы. Однако указанная информация не является упорядоченной в соответствии с существующими требованиями интеллектуального анализа процессов.

Изложенное определяет актуальность проблемы предварительной структуризации журналов регистрации событий в целях построения моделей процессов методами process mining и дальнейшего усовершенствования таких процессов.

2. Анализ научных публикаций

Проблема интеллектуального анализа слабоструктурированных бизнес-процессов рассмотрена в работе [1]. В [1,2] предложены модели неструктурированного и структурированного процессов, а также метод структуризации журналов регистрации событий в задачах интеллектуального анализа процессов. В работе [3] предложен подход к использованию иерархии шаблонов в process mining, который позволяет структурировать слабоструктурированный бизнес-процессы. В [4] рассматривается автоматическое формирование набора ограничений в задаче предварительной обработки ЖРС. В работе [5] рассмотрен подход к верификации бизнес-правил на основе журнала регистрации событий, задающий завершающие действия при предварительной обработке таких журналов. В [6] описывается автоматизированное формирование ограничений на исполнение БП, что дает возможность структурировать журнал регистрации событий на основе набора правил.

Анализ публикаций показывает, что в настоящее время разработаны существенные элементы предварительной структуризации журналов регистрации событий для слабоструктурированных процессов, однако единая технология преобразования таких журналов отсутствует, что и определяет важность и актуальность темы представленной работы.

3. Постановка задачи

Цель предварительной структуризации журналов регистрации событий для слабоструктурированных бизнес-процессов состоит в их преобразовании в такую форму, которая пригодна для их обработки методами интеллектуального анализа сильно структурированных процессов.

Исходными данными задачи являются:

- регистрационный файл слабоструктурированных бизнес-процессов, который содержит в общем случае неупорядоченное описание событий, происходящих на объекте управления с указанием метки времени;
- опциональный перечень объектов, с которыми оперирует бизнес-процесс.

Необходимо разработать технологию, которая бы упорядочивала набор событий журнала с учетом последовательности обработки объектов бизнес-процесса и, тем самым, преобразовывала бы его в форму журнала, аналогичную формируемой ВРМ-системами для сильно структурированных бизнес-процессов. Это позволит использовать существующие методы process mining для анализа и построения моделей слабоструктурированных бизнес-процессов.

4. Технология предварительной структуризации журналов регистрации событий

Для решения задачи структуризации ЖРС предлагается информационная технология предварительной обработки журналов регистрации событий в задачах интеллектуального анализа процессов, которая позволяет преобразовывать неструктурированный ЖРС в структурированную форму, пригодную для обработки инструментальными средствами process mining.

Схема, отражающая основные элементы предлагаемой информационной технологии, представлена на рисунке.

Данная технология включает в себя следующие блоки:

- блок 1 “Извлечение информации о событиях процесса за заданный временной период”;
- блок 2 “Выявление закономерностей наступления событий ЖРС”;
- блок 3 “Структурирование журнала регистрации событий на основе выявленных закономерностей”;
- блок 4 “Верификации структурированного ЖРС”.

Рассмотрим более подробно каждый из блоков технологии предварительной структуризации ЖРС слабоструктурированных бизнес-процессов.

Блок 1. Реализация данного блока состоит из двух этапов:

- извлечение набора событий из ЖРС за заданный промежуток времени;
- проверка структуры полученного набора событий из ЖРС.

Под извлечением набора событий из ЖРС за заданный промежуток времени следует понимать получение данных о событиях процесса из соответствующей БД либо таблицы БД информационной системы, фиксирующей происходящие на объекте управления события.

Информация извлекается для временного интервала $(t_b, t_e) | t_b, t_e \in T$, где t_b – начальная временная отметка; t_e – конечная временная отметка; T – множество всех значений атрибута времени, входящих в ЖРС.

Тогда условие наличия события в выходном наборе имеет следующий вид:

$$(\forall s_{t_i} \in S^*)(t_b \leq t_i \leq t_e), S^* \subset S_A, i = \overline{1, I}, \quad (1)$$

где S^* – отобранный набор событий; S_A – весь ЖРС, хранимый в системе предприятия; s_{t_i} – запись ЖРС; t_i – текущий момент времени.

На втором этапе данного блока осуществляется проверка обязательных атрибутов описания БП, которые необходимы для корректного выполнения технологии структуризации. К обязательным атрибутам слабоструктурированного (неструктурированного) БП относят:

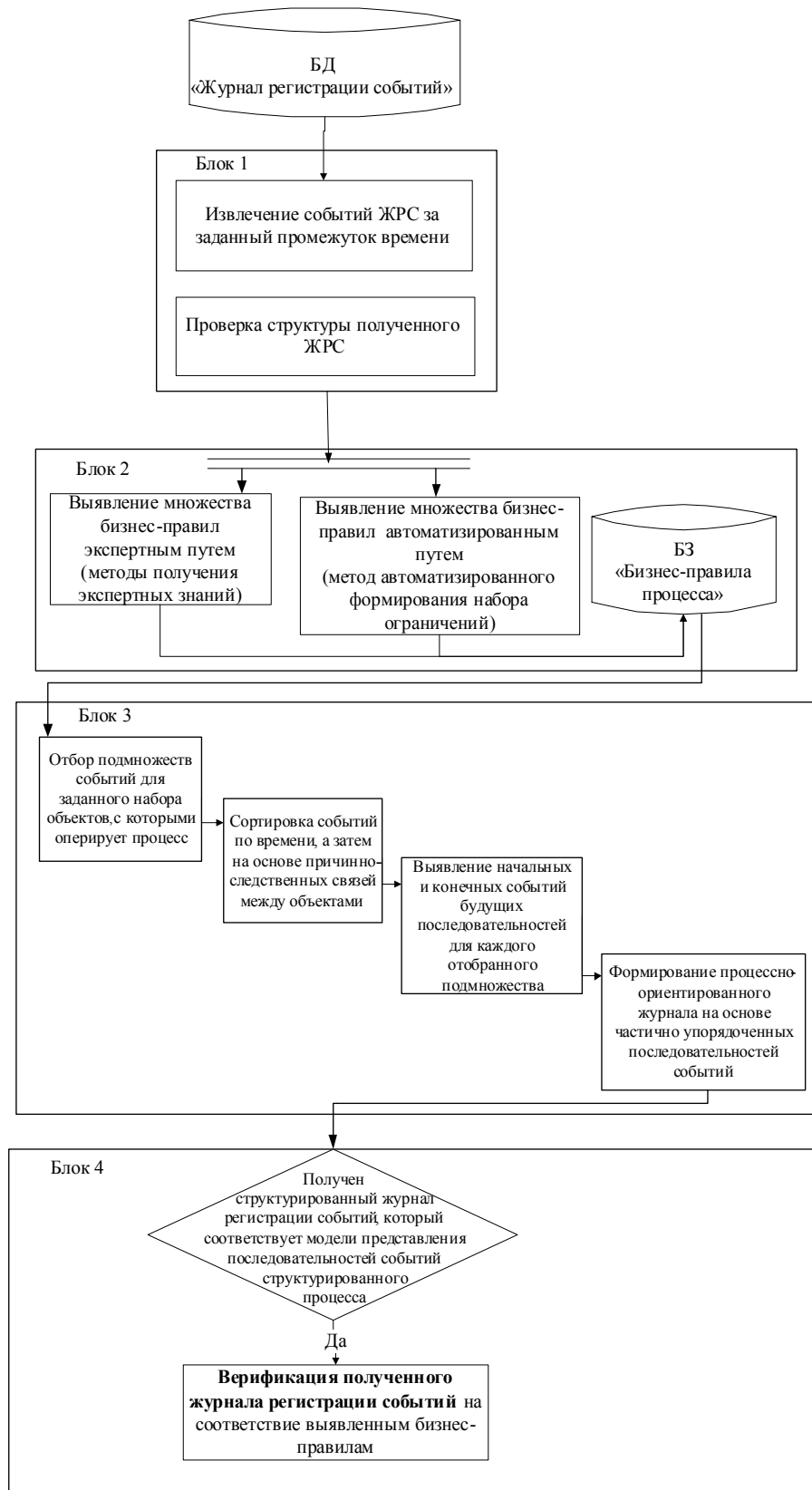
- идентификатор последовательности выполнения процедуры;
- временной параметр;
- действие (событие);
- ресурс (пользователь).

Блок 2 подразумевает формирование бизнес-правил экспертным и/или автоматическим путем. На вход данному блоку подается отобранный набор событий. На выходе блока получаем множество бизнес-правил, ограничивающих выполнение бизнес-процесса.

Формирование бизнес-правил на основе экспертных оценок требует проведения ряда известных организационных мероприятий, в частности интервьюирования, анкетирования, мозгового штурма и т.д.

Для автоматического формирования бизнес-правил был разработан метод формирования набора ограничений БП [6], который подразумевает аналитическую обработку ЖРС БП. Метод формирования подразумевает выполнение следующих этапов:

- группирование событий по объектам;
- подсчет частоты наступления последовательностей событий ЖРС;
- распознавание структурных шаблонов в выявленных последовательностях;
- представление выявленных шаблонов как набора событий.



Технология предварительной структуризации журнала регистрации событий слабоструктурированных бизнес-процессов

Отдельно следует обратить внимание на этап распознавания структурных шаблонов в выявленных закономерностях. На данном этапе осуществляется фильтрация последова-

тельностью событий, учитываемых при анализе, по показателю частоты повторений в ЖРС. В зависимости от заданного порога повторений пользователь получает разные наборы бизнес-правил для анализа. Таким образом, возможен возврат к данному блоку из последующих в случае противоречивости набора полученных правил либо наличия проблем при верификации структурированного ЖРС. Последний, в свою очередь, строится непосредственно на основе бизнес-правил.

Блок 3. В рамках данного блока осуществляется непосредственное формирование ЖРС, структура которого соответствует модели структурированного ЖРС, а сам ЖРС применим для дальнейшего анализа методами интеллектуального анализа процессов.

Для получения данного файла в рамках предлагаемой технологии используется метод структуризации ЖРС. На вход блоку 3 подается полученный в первом блоке ЖРС.

В рамках данного блока выполняются следующие этапы:

- отбор подмножеств событий для заданного набора объектов, с которыми оперирует процесс;
- сортировка событий по времени, а затем на основе причинно-следственных связей между объектами;
- выявление начальных и конечных событий будущих последовательностей для каждого отобранного подмножества;
- формирование процессно-ориентированного журнала на основе частично упорядоченных последовательностей событий.

Первый этап подразумевает группировку событий по заданным объектам БП. На втором этапе осуществляется упорядочивание событий по временному параметру, а также в соответствии с бизнес-правилами, которые были выявлены во втором блоке. На последнем этапе осуществляется выявление входа и выхода БП.

В результате выполнения перечисленных этапов осуществляется упорядочивание множеств последовательностей событий БП по временному параметру. Каждое из указанных множеств отражает обработку связной последовательности объектов на заданном интервале времени и соответствует одному из подпроцессов общего бизнес-процесса.

Блок 4 предполагает проверку полноты и непротиворечивости бизнес-правил, выделенных в блоке 2. Такая верификация должна осуществляться по следующим четырем аспектам бизнес-процесса: последовательность действий процесса; последовательность ситуаций БП; последовательность операций с объектами бизнес-процесса; последовательность взаимодействия процесса с существующей организационной структурой.

Предложенный подход создает условия для адаптации политики предприятия, которая формализуется в виде бизнес-правил, к текущему состоянию организации, а также к неконтролируемым внешним возмущениям.

5. Выводы

Предложена технология предварительной структуризации ЖРС слабоструктурированных бизнес-процессов в задачах интеллектуального анализа процессов. Она включает в себя блоки извлечения информации о событиях процесса за заданный временной период, выявления закономерностей наступления событий ЖРС, структурирования журнала регистрации событий, верификации структурированного ЖРС. Данная технология создает условия для повышения эффективности функционирования слабоструктурированных бизнес-процессов путем выявления несоответствия процесса и его ограничений на основе анализа последовательности событий такого процесса.

В практическом аспекте предложенная технология позволяет расширить сферу применения методов и инструментальных средств process mining на слабоструктурированные и неструктурированные бизнес-процессы, что дает возможность оценить адекватность их моделей и, при необходимости, выполнить реинжиниринг бизнес-процессов.

Список литературы: 1. *Van der Aalst W. M. P. Finding Structure in Unstructured Processes: The Case for Process Mining / W. M. P. van der Aalst // ACSD '07 Proceedings of the Seventh International Conference on Application of Concurrency to System Design. IEEE Computer Society Washington, 2007. С. 3-12.* 2. *Левыкин В. М. Метод предварительной обработки журналов регистрации событий в задачах интеллектуального анализа процессов / В. М. Левыкин, С. Ф. Чалый, Е. О. Богатов, М. И. Дикусар // Международная научно-практическая конференция «Полупроводниковые материалы, информационные технологии и*

фотовольтаника»: сборник тезисов докладов. Кременчуг, 2011. Вып № 7 (88). С. 203–207. **3. Богатов Е.О.** Предварительная обработка журналов регистрации событий в задачах интеллектуального анализа процессов / *Е. О. Богатов, М. И. Дикусар, рук. С.Ф. Чалый* // 16-й Международный молодежный форум «Радиоэлектроника и молодежь в XXI веке»: сборник тезисов докладов. Харьков, 2012. ХНУРЭ. С. 203–207. **4. Чалый С.Ф.** Використання ієрархії шаблонів у process mining/ *С.Ф. Чалый, Є.О. Богатов, Альшейх Алі Джаміль* // Системи обробки інформації. 2010. Вып. №6(87). С. 257–259. **5. Чалый С.Ф.** Верифікація бізнес-правил на основі журналу подій / *С.Ф. Чалый, Є.О. Богатов, І.Б. Буцукина* // Системи обробки інформації. 2010. Вып. №7(88). С. 174–176. **6. Чалый С.Ф.** Формирование набора ограниченный в задаче предварительной обработки неструктурированных журналов регистрации событий / *С. Ф. Чалый, Е. О. Богатов, А. Ю. Кальницкая* // Вісник Національного технічного університету «Харківський політехнічний інститут». 2013. Вып. № 2 (976). С 139 – 148 с.

Поступила в редколлегию 23.11.2012

Чалый Сергей Федорович, д-р техн. наук, профессор кафедры ИУС ХНУРЭ. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-451.

Богатов Евгений Олегович, аспирант кафедры ИУС ХНУРЭ. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-451.

Мелешко Дмитрий Геннадиевич, магистрант кафедры ИУС ХНУРЭ. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-451.

УДК 007.5; 004.85

С.Ф. ЧАЛЫЙ, И.Б. БУЦУКИНА

МОДЕЛЬ БИЗНЕС-ПРОЦЕССА С ИЗМЕНЯЕМОЙ НА ОСНОВЕ ПРАВИЛ СТРУКТУРОЙ

Предлагается модель бизнес-процесса с изменяемой структурой, которая включает в себя наборы бизнес-правил, задающих взаимосвязи между процедурами процесса и ограничивающих выполнение таких процедур. Модель обеспечивает возможности динамической адаптации бизнес-процесса к внешним воздействиям.

1. Актуальность

Современное процессное управление требует построения гибких бизнес-процессов (БП), ориентированных на пользователя и изменяющихся на основе знаний о функционировании предприятия. Такие знания представляются в виде двух составляющих: знания о БП на предприятии, в виде допустимых последовательностей процедур; знания о порядке управления такими процессами, в виде бизнес-правил. Бизнес-правила формализуют отношения (обычно в форме причинно-следственных связей) между процедурами процесса, объектами, с которыми оперирует процесс, а также описывают временные параметры БП. Совокупность таких правил также составляет систему ограничений для БП. Указанная система ограничений задает допустимые траектории реализации БП, допустимые последовательности составляющих процесс процедур, что и определяет возможности их адаптации.

Недостаток гибкости современных моделей бизнес-процессов в значительной степени определяется существующим алгоритмически-ориентированным механизмом описания последовательности работ при выполнении БП. В соответствии с таким механизмом алгоритм функционирования процесса основывается на последовательности предварительно определенных шаблонов, отражающих взаимосвязи между процедурами. Алгоритмический характер описания БП требует жесткого задания последовательности работ на этапе его конфигурирования, что значительно затрудняет проектирование бизнес-процессов с изменяемой структурой. В то же время, необходимость формирования гибких моделей БП с учетом существующих ограничений в форме множества бизнес-правил требует применения подходов, разработанных в области искусственного интеллекта, а именно подходов к решению задач, представленных в виде системы логических ограничений [1]. Изложенное выше определяет актуальность рассматриваемой в статье проблемы моделирования адаптируемых БП с изменяемой структурой с учетом системы логических ограничений в форме бизнес-правил.

фотовольтаника»: сборник тезисов докладов. Кременчуг, 2011. Вып № 7 (88). С. 203–207. **3. Богатов Е.О.** Предварительная обработка журналов регистрации событий в задачах интеллектуального анализа процессов / *Е. О. Богатов, М. И. Дикусар, рук. С.Ф. Чалый* // 16-й Международный молодежный форум «Радиоэлектроника и молодежь в XXI веке»: сборник тезисов докладов. Харьков, 2012. ХНУРЭ. С. 203–207. **4. Чалый С.Ф.** Використання ієрархії шаблонів у process mining/ *С.Ф. Чалый, Є.О. Богатов, Альшейх Алі Джаміль* // Системи обробки інформації. 2010. Вып. №6(87). С. 257–259. **5. Чалый С.Ф.** Верифікація бізнес-правил на основі журналу подій / *С.Ф. Чалый, Є.О. Богатов, І.Б. Буцукина* // Системи обробки інформації. 2010. Вып. №7(88). С. 174–176. **6. Чалый С.Ф.** Формирование набора ограниченный в задаче предварительной обработки неструктурированных журналов регистрации событий / *С. Ф. Чалый, Е. О. Богатов, А. Ю. Кальницкая* // Вісник Національного технічного університету «Харківський політехнічний інститут». 2013. Вып. № 2 (976). С 139 – 148 с.

Поступила в редколлегию 23.11.2012

Чалый Сергей Федорович, д-р техн. наук, профессор кафедры ИУС ХНУРЭ. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-451.

Богатов Евгений Олегович, аспирант кафедры ИУС ХНУРЭ. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-451.

Мелешко Дмитрий Геннадиевич, магистрант кафедры ИУС ХНУРЭ. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-451.

УДК 007.5; 004.85

С.Ф. ЧАЛЫЙ, И.Б. БУЦУКИНА

МОДЕЛЬ БИЗНЕС-ПРОЦЕССА С ИЗМЕНЯЕМОЙ НА ОСНОВЕ ПРАВИЛ СТРУКТУРОЙ

Предлагается модель бизнес-процесса с изменяемой структурой, которая включает в себя наборы бизнес-правил, задающих взаимосвязи между процедурами процесса и ограничивающих выполнение таких процедур. Модель обеспечивает возможности динамической адаптации бизнес-процесса к внешним воздействиям.

1. Актуальность

Современное процессное управление требует построения гибких бизнес-процессов (БП), ориентированных на пользователя и изменяющихся на основе знаний о функционировании предприятия. Такие знания представляются в виде двух составляющих: знания о БП на предприятии, в виде допустимых последовательностей процедур; знания о порядке управления такими процессами, в виде бизнес-правил. Бизнес-правила формализуют отношения (обычно в форме причинно-следственных связей) между процедурами процесса, объектами, с которыми оперирует процесс, а также описывают временные параметры БП. Совокупность таких правил также составляет систему ограничений для БП. Указанная система ограничений задает допустимые траектории реализации БП, допустимые последовательности составляющих процесс процедур, что и определяет возможности их адаптации.

Недостаток гибкости современных моделей бизнес-процессов в значительной степени определяется существующим алгоритмически-ориентированным механизмом описания последовательности работ при выполнении БП. В соответствии с таким механизмом алгоритм функционирования процесса основывается на последовательности предварительно определенных шаблонов, отражающих взаимосвязи между процедурами. Алгоритмический характер описания БП требует жесткого задания последовательности работ на этапе его конфигурирования, что значительно затрудняет проектирование бизнес-процессов с изменяемой структурой. В то же время, необходимость формирования гибких моделей БП с учетом существующих ограничений в форме множества бизнес-правил требует применения подходов, разработанных в области искусственного интеллекта, а именно подходов к решению задач, представленных в виде системы логических ограничений [1]. Изложенное выше определяет актуальность рассматриваемой в статье проблемы моделирования адаптируемых БП с изменяемой структурой с учетом системы логических ограничений в форме бизнес-правил.

2. Анализ научных публикаций

Проблема моделирования гибких адаптируемых БП связана с построением декларативного описания предметной области в виде системы ограничений - связей между объектами, которые управляются с помощью процессов. Они определяют взаимосвязи между процедурами, оперирующими с таким объектами. Это требует рассмотрения подходов, принятых в Constraint programming [1]. Основная идея в рамках данной парадигмы заключается в том, что отношения между переменными предметной области представляются в виде ограничений. Тогда для решения поставленной задачи пользователь должен задать ограничения и получить ответ, удовлетворяющий заданным исходным ограничениям (отношениям), с помощью некоторого универсального решателя. Данный подход позволяет спроектировать гибкие на уровне декларативного описания БП [2,3], а также адаптировать их во время выполнения ограничений. В то же время, при построении моделей БП используется алгоритмический подход, что ограничивает возможность их адаптации с учетом существующих ограничений в форме бизнес-правил. Это определяет важность и актуальность темы данной статьи.

3. Постановка задачи

Исходными данными задачи являются: структурные элементы бизнес-процесса; бизнес-правила, задающие ограничения на процесс. Необходимо разработать модель гибкого бизнес-процесса с изменяемой структурой, обеспечивающего его динамическую адаптацию при изменении условий функционирования. Это позволило бы обеспечить, с одной стороны, сокращение возможных потерь в условиях как эволюционных, так и непредсказуемых изменений внешней среды. С другой стороны, адаптация позволяет лучше использовать возможности, которые возникают в результате внешних изменений, и тем самым повысить эффективность бизнес-процесса.

4. Модель бизнес-процесса с изменяемой структурой

Предлагаемая модель гибкого бизнес-процесса основана на новом подходе к его структуризации. Традиционным является workflow – описание БП, при котором ключевым элементом бизнес-процесса является его последовательность работ (процедур). Такая последовательность может реализовывать достаточно сложные алгоритмы, отражающие взаимодействие как автоматически выполняющихся, так и интерактивных фрагментов процесса. Все возможные варианты выполнения workflow должны быть определены заранее, на этапе создания модели процесса. Следовательно, в традиционную модель процесса неявно включены возможные ограничения, и каждый допустимый вариант выполнения БП прорабатывается с учетом существующих ограничений. В то же время, такие «жесткие» модели бизнес-процессов не позволяют эффективно адаптировать их при изменении внешних условий. Необходимо выполнить их перепроектирование. Структуризация workflow – описания показывает, что элементами бизнес-процесса являются последовательности процедур(activities), сформированные таким образом, чтобы удовлетворять существующим ограничениям в форме бизнес-правил. В то же время бизнес-правила можно рассматривать двояко: как ограничения и как элемент управления. С одной стороны, правила ограничивают выполнение тех или иных процедур в заданных внешних условиях. С другой стороны, бизнес-правила определяют допустимый порядок выполнения процедур – управляют выполнением процесса. Следовательно, workflow целесообразно рассматривать как набор процедур, связи между которыми задаются бизнес-правилами второго типа, а допустимые последовательности этих процедур определяются правилами первого типа. При этом каждая процедура определяется на основе единого объекта (ресурса), с которым она работает, и (или) одного исполнителя, который ее выполняет. Выделение составляющих бизнес-процесса с изменяемой структурой позволяет представить его статическое описание Bp в виде (1) при выполнении необходимых и достаточных условий (2) и (3):

$$Bp = (Pr, Br, Cr, Wf), \quad (1)$$

$$(\forall br_j \in Br) \exists Pr_i \subset Pr \rightarrow br_j(Pr_i) = true, br_j \in wf_m, wf_m \in Wf_1, \quad (2)$$

$$(\forall cr_k \in Cr) \exists Wf_i \subset Wf \rightarrow cr_k(Wf_i) = true, \quad (3)$$

где Pr – множество процедур бизнес-процесса; Bp – множество бизнес-правил, определяющих допустимые последовательности процедур бизнес-процесса; Cr – множество бизнес-правил, ограничивающих выполнение последовательности процедур бизнес-процесса; Wf – множество допустимых последовательностей процедур бизнес-процесса.

Условие (2) задает допустимые взаимосвязи между процедурами, необходимые для построения элементов Wf_i , а условие (3) связано с полнотой модели и задает существование Wf , удовлетворяющих ограничениям Cr . Отметим, что правило $br_j(Pr_i)$ связывает несколько заданных процедур при выполнении определенных в правиле условий, поскольку становится истинным только для его процедур-аргументов.

Предыдущее рассмотрение касалось только статического описания гибкого БП. Выполнение БП описывается последовательностью событий tr_i , каждое из которых отражает выполнение соответствующей процедуры pr_k бизнес-процесса в определенный момент времени τ_i (т.е. фиксирует состояние процедуры). Совокупность этих событий T_j , упорядоченная по времени, образует текущую траекторию выполнения БП:

$$Tr = \{tr_i\}, tr_i = \langle T_1, \dots, T_j, \dots, T_n \rangle, T_j = \{(pr_k, \tau_i)\}, pr_k \in Pr. \quad (4)$$

5. Выводы

Впервые предложена модель бизнес-процесса с изменяемой структурой, которая включает систему ограничений в форме бизнес-правил двух типов, с одной стороны задающих допустимые потоки работ на основе возможных взаимосвязей между процедурами процесса в форме таких правил, а с другой – определяющих недопустимые действия процесса на основе правил-ограничений. Данная модель создает условия для адаптации бизнес-процесса в условиях эволюционных изменений внешней среды, а также непредвиденных внешних воздействий. В практическом аспекте предложенная модель обеспечивает условия для оптимизации бизнес-процесса на основе максимального использования возможностей, предоставляемых правилами первого типа (причинно-следственные зависимости между процедурами), и минимизации ограничений, задаваемых правилами второго типа. Иными словами, модель обеспечивает возможность изменения процесса в целях более эффективного использования ресурсов с учетом заданных ограничений. Второе преимущество предлагаемой модели состоит в том, что адаптация процесса позволяет «расширить узкие места» бизнес-процесса и, тем самым, ослабить существующие ограничения.

Список литературы: 1. Rossi F. Handbook of Constraint Programming / Francesca Rossi, Peter Van Beek, Toby Walsh, Thom Fröhlich, Laurent Michel, Christian Schulte // Amsterdam ; Oxford : Elsevier, 2006. 955 p. 2. Schonenberg M.H. Towards a Taxonomy of Process Flexibility (Extended Version) / M.H. Schonenberg, R.S. Mans, N.C. Russell, N.A. Mulyar, and W.M.P. van der Aalst // BPM Center Report BPM-07-11, BPMcenter.org, 2007. 30 p. 3. Schonenberg M.H. Towards a Taxonomy of Process Flexibility/ M.H. Schonenberg, R.S. Mans, N. Russell, N.A. Mulyar and W.M.P. van der Aalst // In J.L.G. Dietz, A. Albani, J. Barjis (Eds.), Advances in Enterprise Engineering (4th International Workshop CIAO! and 4th International Workshop EOMAS, held at CAiSE 2008, Montpellier, France, June 16-17, 2008, Proceedings). (Lecture Notes in Business Information Systems, Vol. 10. P. 16-30). Berlin: Springer.

Поступила в редколлегию 14.12.2012

Чалый Сергей Федорович, д-р техн. наук, профессор кафедры ИУС ХНУРЭ. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-451.

Буцукина Инна Борисовна, доцент кафедры экономической кибернетики ХНУРЭ. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-451.

СЕРВИС-ОРИЕНТИРОВАННАЯ МОДЕЛЬ БИЗНЕС-ПРОЦЕССА С ИЗМЕНЯЕМОЙ СТРУКТУРОЙ

Предлагается сервис - ориентированная модель бизнес-процесса с изменяемой структурой, представляющая бизнес-процесс в виде набора сервисов, а также набора шаблонов их взаимодействия. Модель позволяет учитывать существующую организационную структуру при построении системы процессов предприятия.

1. Актуальность

Одной из серьезных проблем, которые возникают при построении системы бизнес-процессов и реализации на этой основе процессного подхода к управлению, является несоответствие «горизонтально-ориентированного» построения бизнес-процессов (БП) «вертикально-ориентированной» организационной структуре предприятия, соответствующей функциональному подходу к управлению. Функциональный подход подразумевает наличие нескольких уровней управления от руководителя организации до исполнителя. Подразделения организации составляют иерархическую систему и обычно сгруппированы по видам деятельности. Внутри каждого подразделения существует функциональная иерархия от начальника к исполнителю. Процессный подход предполагает замену вертикального управления структурными подразделениями организации на управление бизнес-процессами, связывающими воедино деятельность этих подразделений. БП горизонтально пронизывают организационную структуру и предполагают различные варианты реализации процессов. Поэтому внедрение процессного подхода существенно влияет на существующую организационную структуру, приводит к наложению «зон ответственности» руководителей подразделений и владельцев процессов, что может затруднить решение задач управления предприятием. Это определяет важность построения и реализации адаптируемых бизнес-процессов, а также актуальность рассматриваемой в статье проблемы согласования БП с существующей организационной структурой на основе построения моделей гибких сервис - ориентированных бизнес-процессов.

2. Анализ научных публикаций

Проведенный анализ показал, что основными особенностями функционального подхода являются [1]: иерархическая структура организации; применение конвейерных технологий; управление организацией по структурным элементам – подразделениям; взаимодействие между подразделениями через структурные элементы более высокого уровня. Рассмотренные в работах [2-4] особенности реорганизации БП и построения процессно-ориентированных систем управления позволяют сделать вывод о том, что процессный подход характеризуется следующими особенностями: организация представляется в виде системы взаимодействующих процессов, направленных на достижение ее глобальной цели; бизнес-процесс реализуется несколькими подразделениями, тогда как выполняемые функции закреплены за конкретным подразделением; взаимодействие процессов выполняется по принципу «производитель – потребитель»; выполняется постоянный мониторинг, измерение и анализ процессов; функционирование процессов выполняется в соответствии с заданными критериями результативности. Сервис-ориентированная архитектура [5] позволяет многократно использовать сервисы, что дает возможность построить гибкие бизнес-процессы с учетом организационной структуры предприятия.

3. Постановка задачи

При решении задачи в качестве исходных данных можно рассматривать структурные элементы бизнес-процесса, включая правила и шаблоны взаимодействия, а также элементы организационной структуры и порядок их взаимодействия. Задача заключается в разработке модели бизнес-процесса на основе иерархии сервисов, взаимодействие между которыми определяется на основе фиксированного набора шаблонов. Последние задают возможные варианты последовательностей сервисов при выполнении бизнес-процесса и,

следовательно, ограничивают множество возможных траекторий его реализации. Такая модель позволила бы выстроить иерархию взаимодействующих бизнес-процессов с привязкой к организационной структуре предприятия и, тем самым, избежать наложения «зон ответственности» руководителей подразделений и владельцев процессов, устранив противоречия при распределении ресурсов.

4. Обобщенная сервис - ориентированная модель бизнес-процесса

Предлагаемая сервис - ориентированная модель бизнес-процесса включает в себя два уровня: модель конфигурирования и модель исполнения. Рассмотрим их назначение и состав. Первая модель (1) является декларативной, содержит полный набор сервисов, а также шаблонов иерархии и взаимодействия, охватывая все возможные варианты реализации бизнес-процесса. Она позволяет оценить поведение процесса извне, проанализировав возможные варианты взаимодействий, а также выполнить верификацию процесса, исследовав его полноту и непротиворечивость. Вторая модель содержит подмножество сервисов и правил взаимодействия и является адаптированной под условия конкретного предприятия исполняемой моделью:

$$Bp = (Sr, Tl, Wf), wf_m(Tl_s) \in WF, tl_j(Sr_i) \in Tl_s, Tl_s \subset Tl, wf_m \in Wf, \quad (1)$$

где Sr – множество сервисов, реализующих бизнес-процесс; Tl – множество шаблонов взаимодействия сервисов, задающих допустимые схемы взаимодействия; Wf – множество возможных последовательностей сервисов, реализующих бизнес-процесс.

Отметим, что допустимую схему взаимодействия на основе шаблона можно рассматривать как логическую функцию от сервисов $tl_j(Sr_i)$, входящих в этот шаблон. Тогда формальное представление сервис - ориентированной модели бизнес-процесса, адаптированной под конкретное предприятие, представляет собой набор логических функций на подмножестве шаблонов взаимодействия сервисов (2), причем каждая workflow – последовательность представляется отдельной функцией на шаблонах и будет истинной на всем подмножестве входящих в нее шаблонов (3):

$$Bp^* = Wf_l(Tl_k), (\forall tl_j \in Tl_k) \quad \exists Sr_i \subset Sr \rightarrow tl_j(Sr_i) = true, \quad (2)$$

$$(\forall wf_m \in Wf_l) \quad \exists Tl_s \subset Tl_k \rightarrow wf_m(Tl_s) = true. \quad (3)$$

Шаблоны взаимодействия сервисов позволяют определить различные типы взаимозависимостей между ними, обеспечивая, с одной стороны, модульный принцип сборки модели конкретного бизнес-процесса, а с другой – определяя набор ограничений между ними. Следовательно, один и тот же набор шаблонов может быть использован для различных бизнес-процессов.

Рассматриваемые шаблоны определяют порядок взаимодействия между двумя, тремя и более сервисами и учитывают временные параметры процесса. Базовые шаблоны взаимодействия между сервисами включают в себя: последовательность; параллельное разбиение; синхронизацию; эксклюзивный выбор; простое слияние. Шаблон последовательности определяет безусловный переход от одного сервиса в рамках единого БП к другому. Шаблон параллельного разбиения (И-разбиения) определяет распараллеливание единого потока управления. Каждый из параллельных потоков выполняется одним или последовательностью сервисов. Шаблон синхронизации (И-объединения) задает слияние параллельных потоков управления в единый поток. Каждый из таких параллельных потоков реализуется через последовательность сервисов. Особенность данного шаблона состоит в том, что каждый параллельный поток должен быть выполнен по одному разу. Шаблон эксклюзивного выбора определяет переход к одной из нескольких возможных ветвей. Иными словами, в зависимости от управляющих данных в точке ветвления согласно данному шаблону мы выбираем только один из нескольких возможных сервисов. Шаблон слияния (И-объединения) описывает слияние нескольких ветвей бизнес-процесса, при этом выполниться должен лишь сервис на одной из входных ветвей.

Рассмотрение данных шаблонов позволяет сделать вывод, что они играют роль сложных правил взаимодействия между сервисами, объединяющих несколько правил вида «ЕСЛИ-ТО». В целом данные шаблоны можно свести к комплексным правилам слияния и разбиения.

5. Выводы

Впервые предложена сервис-ориентированная модель бизнес-процесса с изменяемой структурой, которая включает в себя набор сервисов, реализующих действия процесса, а также набор шаблонов, определяющих возможные связи между сервисами. Данная модель дает возможность отобразить бизнес-процесс на иерархическую организационную структуру предприятия и, тем самым, повысить скорость реинжиниринга и усовершенствования, динамически адаптировав типовые бизнес-процессы для различных организационных структур путем добавления (исключения) правил и сервисов. В практическом аспекте предложенная модель позволяет избежать противоречий при распределении ответственности между руководителями структурных подразделений и владельцами процессов, повышая эффективность их работы.

Список литературы: 1. Репин В.В., Елиферов В.Г. Процессный подход к управлению. Моделирование бизнес-процессов. М.: РИА «Стандарты и качество», 2004. 408 с. (Серия «Практический менеджмент»). 2. Хаммер М., Чампи Дж. Реинжиниринг корпорации: Манифест революции в бизнесе // Пер. с англ.: СПб.: Издательство С.-Петербургского университета, 1997. 332 с. 3. Бондаренко М.Ф., Маторин С.И., Соловьева Е.А. Моделирование и проектирование бизнес-систем: методы, стандарты, технологии. Харьков: Компания СМИТ, 2004. 272 с. 4. W.M.P. van der Aalst. Business Process Management Demystified: A Tutorial on Models, Systems and Standards for Workflow Management. In J. Desel, W. Reisig, and G. Rozenberg, editors, Lectures on Concurrency and Petri Nets, volume 3098 of Lecture Notes in Computer Science, pages 1-65. Springer-Verlag, Berlin, 2004. 5. Thomas Erl. Service-Oriented Architecture: Concepts, Technology & Design. Prentice Hall, 2005. 611 p.

Поступила в редколлегию 21.12.2012

Чалый Сергей Федорович, д-р техн. наук, профессор кафедры ИУС ХНУРЭ. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-451.

АльШейх Али Джамиль, аспирант кафедры ИУС ХНУРЭ. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-451.

УДК 519.7

А.Н. ГВОЗДИНСКИЙ, Н.А. ЯКИМОВА, В.А. ГУБИН

БИНАРНЫЕ ПРЕДИКАТЫ ПРИ ОПИСАНИИ БУЛЕВЫХ ЛОГИЧЕСКИХ ПРОСТРАНСТВ

Устоявшееся представление о математической логике как о науке, изучающей законы мышления с применением аппарата математики, главным образом, для нужд самой математики, в современных условиях становится слишком узким [1]. С расширением областей применения и дальнейшим развитием математической логики изменяется и взгляд на нее. Объектами математической логики являются любые дискретные конечные системы, а ее главная задача – структурное моделирование таких систем. Человеческий язык, как явление дискретное, естественно, должен описываться средствами дискретной математики.

1. Введение

Для описания естественного человеческого языка лучше всего подошел бы аппарат уравнений, подобный аппарату, используемому в математическом анализе, но отличающийся от последнего тем, что он предназначен для формализации не непрерывных, а дискретных процессов. Такой язык дают логические исчисления, а именно: исчисление высказываний и исчисление предикатов. Однако чтобы иметь возможность эффективно решать указанные уравнения, необходимо довести эти исчисления до уровня алгебраической системы [2, с.54]. В классической линейной алгебре широко используется аппарат матриц. Но мы имеем возможность записывать только двухмерные матрицы, не говоря уже о громоздкости этой записи, возрастающей по мере увеличения размерности матриц.

Актуальность исследования. С учетом особенностей логической алгебры представляется возможным разработать метод представления булевых матриц, сокращающий их запись, а также допускающий отсутствие ограничений на их арность. Этот метод изложен в [3].

5. Выводы

Впервые предложена сервис-ориентированная модель бизнес-процесса с изменяемой структурой, которая включает в себя набор сервисов, реализующих действия процесса, а также набор шаблонов, определяющих возможные связи между сервисами. Данная модель дает возможность отобразить бизнес-процесс на иерархическую организационную структуру предприятия и, тем самым, повысить скорость реинжиниринга и усовершенствования, динамически адаптировав типовые бизнес-процессы для различных организационных структур путем добавления (исключения) правил и сервисов. В практическом аспекте предложенная модель позволяет избежать противоречий при распределении ответственности между руководителями структурных подразделений и владельцами процессов, повышая эффективность их работы.

Список литературы: 1. Репин В.В., Елиферов В.Г. Процессный подход к управлению. Моделирование бизнес-процессов. М.: РИА «Стандарты и качество», 2004. 408 с. (Серия «Практический менеджмент»). 2. Хаммер М., Чампи Дж. Реинжиниринг корпорации: Манифест революции в бизнесе // Пер. с англ.: СПб.: Издательство С.-Петербургского университета, 1997. 332 с. 3. Бондаренко М.Ф., Маторин С.И., Соловьева Е.А. Моделирование и проектирование бизнес-систем: методы, стандарты, технологии. Харьков: Компания СМИТ, 2004. 272 с. 4. W.M.P. van der Aalst. Business Process Management Demystified: A Tutorial on Models, Systems and Standards for Workflow Management. In J. Desel, W. Reisig, and G. Rozenberg, editors, Lectures on Concurrency and Petri Nets, volume 3098 of Lecture Notes in Computer Science, pages 1-65. Springer-Verlag, Berlin, 2004. 5. Thomas Erl. Service-Oriented Architecture: Concepts, Technology & Design. Prentice Hall, 2005. 611 p.

Поступила в редколлегию 21.12.2012

Чалый Сергей Федорович, д-р техн. наук, профессор кафедры ИУС ХНУРЭ. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-451.

АльШейх Али Джамиль, аспирант кафедры ИУС ХНУРЭ. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-451.

УДК 519.7

А.Н. ГВОЗДИНСКИЙ, Н.А. ЯКИМОВА, В.А. ГУБИН

БИНАРНЫЕ ПРЕДИКАТЫ ПРИ ОПИСАНИИ БУЛЕВЫХ ЛОГИЧЕСКИХ ПРОСТРАНСТВ

Устоявшееся представление о математической логике как о науке, изучающей законы мышления с применением аппарата математики, главным образом, для нужд самой математики, в современных условиях становится слишком узким [1]. С расширением областей применения и дальнейшим развитием математической логики изменяется и взгляд на нее. Объектами математической логики являются любые дискретные конечные системы, а ее главная задача – структурное моделирование таких систем. Человеческий язык, как явление дискретное, естественно, должен описываться средствами дискретной математики.

1. Введение

Для описания естественного человеческого языка лучше всего подошел бы аппарат уравнений, подобный аппарату, используемому в математическом анализе, но отличающийся от последнего тем, что он предназначен для формализации не непрерывных, а дискретных процессов. Такой язык дают логические исчисления, а именно: исчисление высказываний и исчисление предикатов. Однако чтобы иметь возможность эффективно решать указанные уравнения, необходимо довести эти исчисления до уровня алгебраической системы [2, с.54]. В классической линейной алгебре широко используется аппарат матриц. Но мы имеем возможность записывать только двухмерные матрицы, не говоря уже о громоздкости этой записи, возрастающей по мере увеличения размерности матриц.

Актуальность исследования. С учетом особенностей логической алгебры представляется возможным разработать метод представления булевых матриц, сокращающий их запись, а также допускающий отсутствие ограничений на их арность. Этот метод изложен в [3].

Цель исследования: графическое представление логических пространств с использованием аппарата бинарных предикатов.

Задача: сопоставление методов описания логических матриц и логических пространств.

2. Методы решения

Рассмотрим предикат, заданный k -мерной матрицей $m_1 \times m_2 \times \dots \times m_k$, определенной на декартовом произведении $G_1 \times G_2 \times \dots \times G_k$, где $G_i = \{g_i^j, j=1, \overline{m_i}\}$, $i=1, k$. Этот предикат задает некоторое подпространство универсума U размерности $\sum_{i=1}^k m_i$. Множество

базисных векторов распадается на k непересекающихся подмножеств (*сортов переменных*) $G_1, G_2, \dots, G_k$, причем только на их декартовом произведении предикат, задающий пространство, определен. Другими словами, это подпространство не содержит точек, определенных на декартовом произведении $G_i \times G_i$ за исключением точек, лежащих на координатных осях. Количество сортов переменных k называется *порядком* описанного выше логического пространства, а само подпространство называется *предикатным логическим пространством порядка k* . Далее будем рассматривать пространства порядка $k=2$. Предикаты, задающие их, в этом случае задаются матрицами размерности $m \times n$, определенными на декартовом произведении $X \times Y$, где $X = \{x_i, i=1, \overline{m}\}$, $Y = \{y_j, j=1, \overline{n}\}$. Размерность такого пространства равна $m+n$. В этом случае его координатные оси можно расположить на конусной поверхности второго порядка, имеющей в декартовой системе координат трехмерного вещественного пространства уравнение следующего вида

$$[1]: \frac{x^2}{a_1^2} + \frac{y^2}{a_2^2} + \frac{z^2}{a_3^2} = 0.$$

На рис. 1 показано предикатное логическое пространство второго порядка L_A , или *бинарнопредикатное логическое пространство*, заданное бинарным предикатом,

представленным матрицей $A_{3 \times 3}: A = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \end{pmatrix}.$

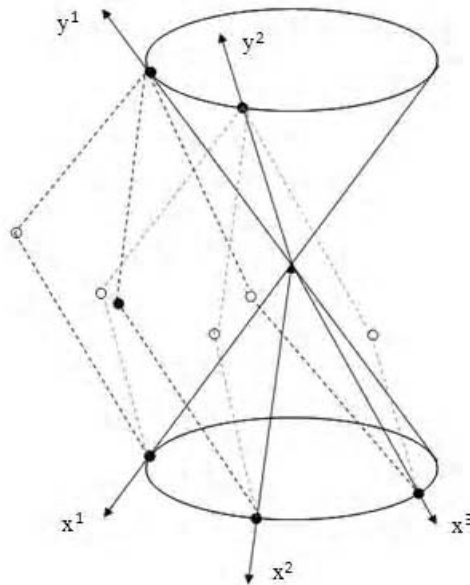


Рис. 1. Бинарнопредикатное логическое пространство L_A

Точки этого пространства окрашены в белый цвет, если в них значение предиката равно единице, и в черный – если значение предиката в них равно нулю [2].

Отрицанию матрицы A

$$\bar{A} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

будет соответствовать подпространство $L_{\bar{A}}$, в котором точки, не лежащие на координатных осях, поменяют окраску на противоположную. При этом точки, принадлежащие координатным осям, не поменяют цвет. Подпространство $L_{\bar{A}}$ называется *отрицанием подпространства* L_A .

Для предиката, представленного дизъюнкцией матриц $A \vee B$, отвечающее ему подпространство будет следующим. Если хотя бы в одном из пространств L_A или L_B соответствующая точка была белой (т.е. если хотя бы один из соответствующих дизъюнктурируемых элементов матриц A и B был равен единице), то и в пространстве $L_{A \vee B}$ она будет также белой. В противном случае (т.е. если оба дизъюнкта были равны нулю) она останется черной. Такое пространство $L_{A \vee B}$ называется *пересечением подпространств* L_A и L_B *по нулю* и обозначается как $L_A \bigcap_0 L_B$.

Для предиката, представленного конъюнкцией матриц $A \wedge B$, отвечающее ему подпространство будет следующим. Точка подпространства $L_{A \wedge B}$ будет белой, если и в пространстве L_A , и в пространстве L_B соответствующие ей точки были белыми (т.е. если оба соответствующие конъюнктурируемые элементы матриц A и B были равны единице). В противном случае (т.е. если хотя бы один из конъюнктов был равен нулю) она останется черной. Такое пространство $L_{A \wedge B}$ называется *пересечением подпространств* L_A и L_B *по единице* и обозначается как $L_A \bigcap_1 L_B$. Имеют место следующие соотношения:

$$(L_A \bigcap_1 L_B) \bigcap_0 L_A = L_A, \quad (1)$$

$$(L_A \bigcap_0 L_B) \bigcap_1 L_A = L_A, \quad (2)$$

$$L_{\overline{A \vee B}} = L_{\bar{A}} \bigcap_1 L_{\bar{B}}, \quad (3)$$

$$L_{\overline{A \wedge B}} = L_{\bar{A}} \bigcap_0 L_{\bar{B}}, \quad (4)$$

$$L_{\bar{A}} \bigcap_0 L_A = L_1, \quad (5)$$

где L_1 – пространство, в котором все точки, не принадлежащие координатным осям, являются белыми;

$$L_{\bar{A}} \bigcap_1 L_A = L_0, \quad (6)$$

здесь L_0 – пространство, все точки которого являются черными:

$$L_A \bigcap_0 L_B \bigcap_1 L_{\bar{B}} = L_A, \quad (7)$$

$$L_A \bigcap_1 (L_B \bigcap_0 L_{\bar{B}}) = L_A, \quad (8)$$

$$L_A = L_A, \quad (9)$$

$$L_A \bigcap_0 L_0 = L_A, \quad (10)$$

$$L_A \bigcap_1 L_0 = L_0. \quad (11)$$

Выражения (1) – (11) имеют смысл только в том случае, когда предикаты, задающие пространства L_A и L_B , имеют одинаковую размерность, причем мощности множеств X и множеств Y соответственно для этих пространств одинаковы.

Рассмотрим теперь пространство L_B , у которого мощность сорта Y произвольная, а мощность сорта X совпадает с мощностью сорта Y пространства L_A . Другими словами, если предикат, задающий пространство L_A , представлен матрицей $A_{m \times n}$, то предикат, задающий пространство L_B , представлен матрицей $B_{n \times p}$. Точки пространства, заданного некоторым предикатом, не принадлежащие координатным осям, назовем *исходящими из* (или *входящими в*) y_i^j , если они принадлежат гиперплоскости, одной из составляющих которой является координатная ось $(0, g_i^j)$. Допустим, что предикат, задающий пространство L_B , задан матрицей

$$B = \begin{pmatrix} 0 & 1 \\ 1 & 1 \\ 1 & 0 \end{pmatrix}.$$

Пространство L_B изображено на рис. 2.

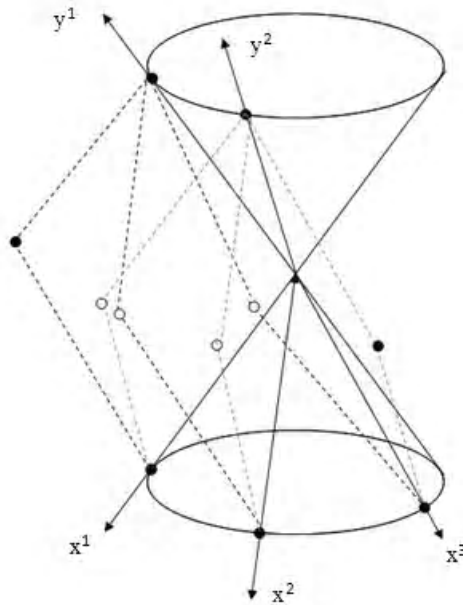


Рис. 2. Бинарнопредикативное логическое пространство L_B

Теперь построим пространство L_C следующим образом. Мощность сорта X пространства L_C будет совпадать с мощностью сорта X пространства L_A , а мощность сорта Y – с мощностью сорта Y пространства L_B . Рассмотрим точки, исходящие из x_i простран-

ства L_A , и точки, входящие в U_j пространства L_B . Обозначим их через $(x^i, y^j)_A$ и $(x^i, y^j)_B$. Разобьем их на пары следующим образом: $\{(x^i, y^1)_A, (x^1, y^j)_B\}, \dots, \{(x^i, y^n)_A, (x^n, y^j)_B\}$. Если хотя бы в одной из этих пар обе точки имеют белый цвет, то точка $(x^i, y^j)_C$ также будет белой. В противном случае соответствующая точка $(x^i, y^j)_C$ будет черной. Построенное таким образом пространство L_C изображено на рис. 3. Ему соответствует предикат, представленный матрицей

$$C = AB = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} 0 & 1 \\ 1 & 1 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \\ 1 & 1 \end{pmatrix}.$$

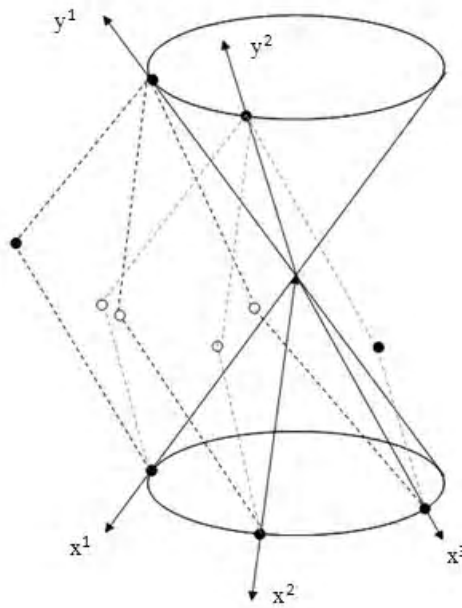


Рис. 3. Бинарнопредикативное логическое пространство L_C

Операция построения пространства L_C называется *наложением пространства L_A на пространство L_B* и обозначается $L_A * L_B$. Свойство коммутативности для этой операции не имеет места. Другими словами, при наложении пространства L_B на пространство L_A результат будет либо другим, либо операция вообще не будет иметь смысла (в случае, когда мощности сортов X и Y пространств L_A и L_B не удовлетворяют сформулированному выше условию). Имея это в виду, можно записать следующие равенства:

$$(L_A \bigcap_0 L_B) * L_C = L_A * L_C \bigcap_0 L_B * L_C;$$

$$L_C * (L_A \bigcap_0 L_B) = L_C * L_A \bigcap_0 L_C * L_B;$$

$$L_A * (L_B * L_C) = (L_A * L_B) * L_C;$$

$$L_A * L_0 = L_0.$$

Таким образом, по аналогии с бинарными предикатами, используя математический аппарат n -арных предикатов, можно записывать и представлять в графическом виде

булевы логические пространства любой размерности и любого порядка. Более того, такое представление упрощает процедуру их программной обработки, так как сокращает объем необходимой памяти.

Выводы

В ходе исследований, изложенных в статье, была установлена возможность графической интерпретации логических пространств по аналогии с бинарными предикатами.

Научная новизна: по результатам проведенного исследования доказана возможность описания булевых логических пространств по аналогии с аппаратом n -арных предикатов.

Практическая значимость: рассмотренный в статье аппарат описания булевых логических пространств может упростить их программную реализацию.

Список литературы: 1. Сигорский В.П. Математический аппарат инженера. Киев: Техника, 1975. 768с. 2. Шабанов-Кушнарченко Ю.П. Теория интеллекта. Математические средства. Харьков: Вища школа. 144с. 3. Гвоздинский А.Н., Якимова Н.А., Губин В.А. Представление булевых логических матриц в виде бинарных предикатов // Радиоэлектроника и информатика. 2007. Вып 2. С. 108 – 110.

Поступила в редколлегию 17.12.2012

Гвоздинский Анатолий Николаевич, канд. техн. наук, профессор кафедры искусственного интеллекта ХНУРЕ. Научные интересы: оптимизация процедур принятия решений в сложных системах управления. Адрес: Украина, 61166, Харьков, ул. акад. Ляпунова, 7, кв. 9, тел. 702-38-23.

Якимова Наталья Анатольевна, д-р техн. наук, профессор кафедры компьютерной алгебры и дискретной математики Одесского национального университета им. И.И. Мечникова. Научные интересы: логическая алгебра, искусственный интеллект. Украина, 65001, Одесса, ул. Дворянская, 2, тел. 8(048)7238405.

Губин Вадим Александрович, ст. преподаватель кафедры искусственного интеллекта ХНУРЕ. Научные интересы: интеллектуальный анализ текстовых данных. Адрес: Украина, 61054, Харьков, ул. Гв. Широнинцев, 23, кв.286, тел. 095-4630201.

УДК 519.21 : 004.77

Е.С. ИЕВЛЕВ

О ВЫБОРЕ ЗАКОНА РАСПРЕДЕЛЕНИЯ ПРОДОЛЖИТЕЛЬНОСТИ ПЕРЕДАЧИ ПАКЕТИРОВАННЫХ ДАННЫХ В КОРПОРАТИВНЫХ КОМПЬЮТЕРНЫХ СЕТЯХ

В связи со стремительным развитием компьютерных технологий одной из наиболее актуальных становится задача разработки моделей управления сетевыми процессами. Сложность ее решения состоит в том, что сетевые процессы в современных компьютерных сетях имеют случайный характер. Анализ результатов многочисленных экспериментов по исследованию сетевых процессов показывает, что переход к технологии пакетной коммутации и создание интегрированных информационных приложений сопровождается сложными явлениями, исследование которых может быть проведено в рамках теоретико-вероятностных подходов [1].

Целью данной работы является выбор закона распределения продолжительности передачи пакетированных данных в корпоративных компьютерных сетях (ККС) для построения вероятностных моделей управления сетевыми процессами.

Во всех корпоративных компьютерных сетях, в которых передача данных (пакетов) подвержена влиянию случайных воздействий, принимается, что продолжительность такой передачи является случайной величиной [1]. Предполагается, что случайные величины продолжительности (времени) передачи пакетов подчинены принятому для данной ККС закону распределения, причем его тип принимается одинаковым для всех передач. Что касается параметров распределения, то последние задаются для каждой передачи на основе либо нормативных данных, либо априорных соображений, либо из статистического опыта.

булевы логические пространства любой размерности и любого порядка. Более того, такое представление упрощает процедуру их программной обработки, так как сокращает объем необходимой памяти.

Выводы

В ходе исследований, изложенных в статье, была установлена возможность графической интерпретации логических пространств по аналогии с бинарными предикатами.

Научная новизна: по результатам проведенного исследования доказана возможность описания булевых логических пространств по аналогии с аппаратом n -арных предикатов.

Практическая значимость: рассмотренный в статье аппарат описания булевых логических пространств может упростить их программную реализацию.

Список литературы: 1. Сигорский В.П. Математический аппарат инженера. Киев: Техника, 1975. 768с. 2. Шабанов-Кушнарченко Ю.П. Теория интеллекта. Математические средства. Харьков: Вища школа. 144с. 3. Гвоздинский А.Н., Якимова Н.А., Губин В.А. Представление булевых логических матриц в виде бинарных предикатов // Радиоэлектроника и информатика. 2007. Вып 2. С. 108 – 110.

Поступила в редколлегию 17.12.2012

Гвоздинский Анатолий Николаевич, канд. техн. наук, профессор кафедры искусственного интеллекта ХНУРЕ. Научные интересы: оптимизация процедур принятия решений в сложных системах управления. Адрес: Украина, 61166, Харьков, ул. акад. Ляпунова, 7, кв. 9, тел. 702-38-23.

Якимова Наталья Анатольевна, д-р техн. наук, профессор кафедры компьютерной алгебры и дискретной математики Одесского национального университета им. И.И. Мечникова. Научные интересы: логическая алгебра, искусственный интеллект. Украина, 65001, Одесса, ул. Дворянская, 2, тел. 8(048)7238405.

Губин Вадим Александрович, ст. преподаватель кафедры искусственного интеллекта ХНУРЕ. Научные интересы: интеллектуальный анализ текстовых данных. Адрес: Украина, 61054, Харьков, ул. Гв. Широнинцев, 23, кв.286, тел. 095-4630201.

УДК 519.21 : 004.77

Е.С. ИЕВЛЕВ

О ВЫБОРЕ ЗАКОНА РАСПРЕДЕЛЕНИЯ ПРОДОЛЖИТЕЛЬНОСТИ ПЕРЕДАЧИ ПАКЕТИРОВАННЫХ ДАННЫХ В КОРПОРАТИВНЫХ КОМПЬЮТЕРНЫХ СЕТЯХ

В связи со стремительным развитием компьютерных технологий одной из наиболее актуальных становится задача разработки моделей управления сетевыми процессами. Сложность ее решения состоит в том, что сетевые процессы в современных компьютерных сетях имеют случайный характер. Анализ результатов многочисленных экспериментов по исследованию сетевых процессов показывает, что переход к технологии пакетной коммутации и создание интегрированных информационных приложений сопровождается сложными явлениями, исследование которых может быть проведено в рамках теоретико-вероятностных подходов [1].

Целью данной работы является выбор закона распределения продолжительности передачи пакетированных данных в корпоративных компьютерных сетях (ККС) для построения вероятностных моделей управления сетевыми процессами.

Во всех корпоративных компьютерных сетях, в которых передача данных (пакетов) подвержена влиянию случайных воздействий, принимается, что продолжительность такой передачи является случайной величиной [1]. Предполагается, что случайные величины продолжительности (времени) передачи пакетов подчинены принятому для данной ККС закону распределения, причем его тип принимается одинаковым для всех передач. Что касается параметров распределения, то последние задаются для каждой передачи на основе либо нормативных данных, либо априорных соображений, либо из статистического опыта.

В системах ККП, например, можно задать три параметра: нижняя грань области определения a (оптимистическое время передачи пакета), верхняя грань b (пессимистическое время передачи пакета) и мода распределения m (наиболее вероятное время передачи пакета). Практически для всех систем ККС априорно можно принять, что плотность распределения временных оценок продолжительности передачи пакетов обладает тремя свойствами: непрерывностью, унимодальностью и двумя неотрицательными точками пересечения этой плотности с осью абсцисс. Простейшим распределением с подобными свойствами является бета-распределение. Общий его вид характеризуется, помимо наличия большого количества случайных факторов, каждый из которых в отдельности оказывает незначительное, несущественное влияние, наличием нескольких, также случайных факторов, число которых невелико, а влияние существенно. В результате воздействия существенных факторов распределение вероятностей обычно делается асимметричным. Отсюда вытекает возможность выбора бета-распределения в качестве априорно типового. Анализ статистических данных (хронометражи продолжительности передачи пакетированных данных) также подтверждают возможность использования бета-распределения в качестве априорного.

Формула плотности бета-распределения имеет следующий вид:

$$B(p, q, x) = \begin{cases} \frac{1}{B(p, q)} x^{p-1} (1-x)^{q-1} & \text{при } 0 \leq x \leq 1, \\ 0 & \text{при } x < 0, x > 1, \end{cases}, \quad (1)$$

где $B(p, q)$ – бета-функция, причем

$$B(p, q) = \int_0^1 x^{p-1} (1-x)^{q-1} dx = \frac{\tilde{A}(p)\tilde{A}(q)}{\tilde{A}(p+q)},$$

а гамма-функция $\tilde{A}(z)$ определяется по формуле

$$\tilde{A}(z) = \int_0^z e^{-t} t^{z-1} dt,$$

причем для целых z функция $\tilde{A}(z) = 1 \cdot 2 \cdot \dots \cdot (z-1) = (z-1)!$

Вид функции (1) зависит от показателей p и q , причем для $p > 2$ (и, соответственно, для $q > 2$) функция распределения обращается в 0 в левой (или правой) конечной точке вместе с ее первой производной. Для $1 < p < 2$ (и, соответственно, $1 < q < 2$) кривая имеет вертикальную касательную в левой (правой) конечной точке. Для $0 < p < 1$ (и, соответственно, $0 < q < 1$) функция уходит в бесконечность, если значения x соответствуют левой (правой) конечной точке, причем вертикальная прямая, проведенная из левой крайней точки, будет ее асимптотой. Для $p \leq 0$ (и, соответственно, $q \leq 0$) интеграл равен бесконечности, так что функция распределения перестает существовать.

Рассмотрим одно из обоснований целесообразности принятия закона бета-распределения, основанное на построении модели случайной величины времени окончания передачи пакетированных данных в ККП.

Пусть начало передачи пакета T_0 , а окончание представляет собой случайную величину, изменяющуюся в интервале (T_1, T_2) .

Плотность распределения случайной величины окончания передачи пакета определим, исходя из следующих предположений:

1. Весь интервал времени передачи пакета (T_0, τ) состоит из интервалов, относящихся к передаче, и интервалов, относящихся к задержкам.
2. Длительность времени, равная $T_1 - T_0$, относится к передаче, а длительность $\tau - T_1$ – к задержкам.
3. Отрезок времени $T_1 - T_0$ разбит на n одинаковых частей длительностью $(T_1 - T_0)/n$.

Если на первом интервале $\left(T_0, T_0 + \frac{T_1 - T_0}{n}\right)$ возникает задержка, то после момента

$t_1 = T_0 + \frac{T_1 - T_0}{n}$ передача пакета прекращается, и в последующий интервал времени от t_1 до $t'_1 = t_1 + \Delta$, где $\Delta = (T_2 - T_1)/n$, возникшая задержка устраняется и передача снова возобновляется только с момента t'_1 . Если на интервале (T_0, t_1) не возникает задержек, то после t_1 передача пакета продолжается. Затем учитывается возможность затруднений на следующем этапе передачи $\left(t'_1, t'_1 + \frac{T_1 - T_0}{n}\right)$ в первом случае и (t_1, t_2) , где $t_2 = t_1 + \frac{T_1 - T_0}{n}$ – во втором и т.д. Очевидно, что если на каждом этапе задержек не возникает, то передача пакета закончится в T_1 , а если задержки возникают на каждом этапе, то передача пакета окончится в момент T_2 . Если в общей сложности возникает m задержек, то передача пакета закончится в момент $\tau = T + m\Delta = T_1 + m \frac{T_2 - T_1}{n}$.

4. Событие, заключающееся в том, что на i -м этапе возникла задержка, определяется i -й выборкой из некоторой генеральной совокупности.

5. Единичный элемент генеральной совокупности содержит долю p «благоприятствуя задержкам».

6. С каждым этапом генеральная совокупность увеличивается на ϑ , причем, если на предыдущем этапе возникли задержки, то ϑ благоприятствовало им и не благоприятствовало в противном случае. Если через A_i^k обозначить событие, заключающееся в том, что на $(i+1)$ -м этапе возникла задержка при условии, что на предыдущих i этапах возникло k задержек, то вероятность события A_i^k будет иметь вид [2]:

$$P(A_i^k) = \frac{p + k\vartheta}{1 + i\vartheta} \quad (1 \leq k \leq i \leq n).$$

7. Разность вероятностей задержек на i -м этапе при наличии $k+1$ и k задержек на предыдущих этапах относится к вероятности задержек на i -м этапе при полном их отсутствии на предыдущих. Эта разность описывается соотношением

$$\frac{P(A_i^{k+1}) - P(A_i^k)}{P(A_i^0)} = \frac{\vartheta}{p}.$$

Из этой формулы видно, что рассматривается такой закон задержек, для которого относительная их величина постоянна. При этом можно показать, что распределение вероятностей для случайной величины m имеет вид:

$$P_{m,n} = C_n^m \frac{\prod_{i=0}^{m-1} (p + i\vartheta) \prod_{i=0}^{n-m-1} (1 - p + i\vartheta)}{\prod_{i=1}^{n-1} (p + i\vartheta)} \quad (1 \leq k \leq i \leq n). \quad (2)$$

Отметим, что отсутствие зависимости вероятности задержек от предыдущего этапа является частным случаем ($\vartheta = 0$) написанного выражения для $p_{m,n}$ и представляет собой известное биномиальное распределение. Далее находится предельное выражение для вероятности $p_{m,n}$ при условии, что n неограниченно возрастает. Из формулы (2) получим

$$\frac{p_{m-1,n}}{p_{m,n}} = \frac{n-m}{m+1} \cdot \frac{p + m\vartheta}{1 - p + (n-m-1)\vartheta}, \quad (3)$$

откуда, обозначив $\frac{p}{\vartheta} = \alpha$, $\frac{p}{\vartheta} \left(\frac{1}{p} - 1 \right) = \beta$, будем иметь

$$\frac{p_{m-1,n} - p_{m,n}}{p_{m,n}} = \frac{(\alpha-1)n + (2-\alpha-\beta)m - \beta + 1}{(m+1)(\beta + n - m - 1)} = \frac{(\alpha-1) + (2-\alpha-\beta)\frac{m}{n} + \frac{1-\beta}{n}}{\frac{m}{n} \left(1 + \frac{m+1}{n} + \frac{\beta}{n} \right)}.$$

Полагая $\frac{m}{n} = x$, $\frac{m+1}{n} = x + \Delta x$, $p_{m,n} = y$, $p_{m+1,n} = y + \Delta y$, устремляя $n \rightarrow \infty$ или $\Delta x \rightarrow 0$ и интегрируя получаем $y = Cx^{n-1}(1-x)^{\beta-1}$, откуда видно, что плотность вероятности случайной величины $\xi = \lim_{n \rightarrow \infty} \frac{m}{n}$, выражается формулой

$$P_{\xi}(x) = \frac{1}{B(\alpha, \beta)} x^{\alpha-1} (1-x)^{\beta-1},$$

в которой $B(\alpha, \beta)$ – функция Эйлера, совпадающая с (1).

Следовательно, ξ является случайной величиной, распределенной по закону бета-распределения (1). Замена переменных $x = (t-a)^{\alpha-1}(b-a)$ приводит к известной формуле бета-распределения с плотностью

$$f(t) = \begin{cases} \frac{1}{(b-a)^{\alpha+\beta-1} B(\alpha, \beta)} (t-a)^{\alpha-1} (b-t)^{\beta-1} & \text{при } a \leq t \leq b, \\ 0 & \text{в противном случае.} \end{cases}$$

Таким образом, в данной статье обоснован выбор бета-распределения в качестве априорно-типичного для описания продолжительности передачи пакетированных данных в корпоративных компьютерных сетях.

Список литературы: 1. *Городецкий А.Я., Заборовский В.С.* Фрактальные процессы в компьютерных сетях: Учебное пособие. СПб.: Изд-во СПбГТУ, 2000. 102 с. 2. *Голенко Д.И.* Моделирование и статистический анализ псевдослучайных величин на электронных вычислительных машинах. М.: Наука, 1965. 228 с.

Поступила в редколлегию 12.12.2012

Иевлев Евгений Сергеевич, аспирант кафедры программной инженерии ХНУРЭ. Научные интересы: математическое моделирование. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел.: (057) 7021-640, e-mail: lmd@kture.kharkov.ua.

УДК 681.3:519.2

Л.О. КИРИЧЕНКО, Т.А. РАДИВИЛОВА, Э. КАЙАЛИ

РАСЧЕТ СТОИМОСТИ МАРШРУТИЗАЦИИ В СЕТИ MPLS С УЧЕТОМ ФРАКТАЛЬНЫХ СВОЙСТВ ТРАФИКА

Описывается метод расчета стоимости маршрутизации сети MPLS, который позволяет минимизировать стоимость маршрутизации с учетом фрактальных свойств трафика, выбора пути его передачи и требований качества обслуживания. Метод использует значения показателя Херста и величину нормированного разброса значений трафика, что дает возможность применять его к самоподобным и мультифрактальным потокам данных.

Введение

Экспериментальные и численные исследования, проведенные в последние десятилетия, свидетельствуют, что трафик во многих мультимедийных сетях обладает фрактальными свойствами. Такой трафик имеет особую структуру, сохраняющуюся на многих масштабах – в реализации всегда присутствует некоторое количество очень больших выбросов при относительно небольшом среднем уровне трафика. Эти выбросы вызывают значительные задержки и потери пакетов, даже когда суммарная потребность всех потоков далека от максимально допустимых значений. Причина такого эффекта заключается в особенностях распределения файлов по серверам, их размерах, в типичном поведении пользователей и в значительной степени связана с изменениями сетевых ресурсов и топологии сети. Оказалось, что изначально не проявляющие свойств самоподобия потоки данных, пройдя обработку на узловых серверах и активных сетевых элементах, начинают подавать ярко

Полагая $\frac{m}{n} = x$, $\frac{m+1}{n} = x + \Delta x$, $p_{m,n} = y$, $p_{m+1,n} = y + \Delta y$, устремляя $n \rightarrow \infty$ или $\Delta x \rightarrow 0$ и интегрируя получаем $y = Cx^{n-1}(1-x)^{\beta-1}$, откуда видно, что плотность вероятности случайной величины $\xi = \lim_{n \rightarrow \infty} \frac{m}{n}$, выражается формулой

$$P_{\xi}(x) = \frac{1}{B(\alpha, \beta)} x^{\alpha-1} (1-x)^{\beta-1},$$

в которой $B(\alpha, \beta)$ – функция Эйлера, совпадающая с (1).

Следовательно, ξ является случайной величиной, распределенной по закону бета-распределения (1). Замена переменных $x = (t-a)^{\alpha-1}(b-a)$ приводит к известной формуле бета-распределения с плотностью

$$f(t) = \begin{cases} \frac{1}{(b-a)^{\alpha+\beta-1} B(\alpha, \beta)} (t-a)^{\alpha-1} (b-t)^{\beta-1} & \text{при } a \leq t \leq b, \\ 0 & \text{в противном случае.} \end{cases}$$

Таким образом, в данной статье обоснован выбор бета-распределения в качестве априорно-типичного для описания продолжительности передачи пакетированных данных в корпоративных компьютерных сетях.

Список литературы: 1. *Городецкий А.Я., Заборовский В.С.* Фрактальные процессы в компьютерных сетях: Учебное пособие. СПб.: Изд-во СПбГТУ, 2000. 102 с. 2. *Голенко Д.И.* Моделирование и статистический анализ псевдослучайных величин на электронных вычислительных машинах. М.: Наука, 1965. 228 с.

Поступила в редколлегию 12.12.2012

Иевлев Евгений Сергеевич, аспирант кафедры программной инженерии ХНУРЭ. Научные интересы: математическое моделирование. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел.: (057) 7021-640, e-mail: lmd@kture.kharkov.ua.

УДК 681.3:519.2

Л.О. КИРИЧЕНКО, Т.А. РАДИВИЛОВА, Э. КАЙАЛИ

РАСЧЕТ СТОИМОСТИ МАРШРУТИЗАЦИИ В СЕТИ MPLS С УЧЕТОМ ФРАКТАЛЬНЫХ СВОЙСТВ ТРАФИКА

Описывается метод расчета стоимости маршрутизации сети MPLS, который позволяет минимизировать стоимость маршрутизации с учетом фрактальных свойств трафика, выбора пути его передачи и требований качества обслуживания. Метод использует значения показателя Херста и величину нормированного разброса значений трафика, что дает возможность применять его к самоподобным и мультифрактальным потокам данных.

Введение

Экспериментальные и численные исследования, проведенные в последние десятилетия, свидетельствуют, что трафик во многих мультимедийных сетях обладает фрактальными свойствами. Такой трафик имеет особую структуру, сохраняющуюся на многих масштабах – в реализации всегда присутствует некоторое количество очень больших выбросов при относительно небольшом среднем уровне трафика. Эти выбросы вызывают значительные задержки и потери пакетов, даже когда суммарная потребность всех потоков далека от максимально допустимых значений. Причина такого эффекта заключается в особенностях распределения файлов по серверам, их размерах, в типичном поведении пользователей и в значительной степени связана с изменениями сетевых ресурсов и топологии сети. Оказалось, что изначально не проявляющие свойств самоподобия потоки данных, пройдя обработку на узловых серверах и активных сетевых элементах, начинают подавать ярко

выраженные признаки самоподобия. Кроме того, этот механизм в значительной степени связан с изменениями сетевых ресурсов и топологии сети [1,2].

В настоящее время эффективным решением задачи обеспечения качества обслуживания сетей является технология многопротокольной коммутации пакетов MPLS (Multiprotocol Label Switching Protocol). Управление трафиком на основе дифференцированного обслуживания в сетях MPLS обеспечивает их масштабируемость с несколькими классами обслуживания, резервирование ресурсов, отказоустойчивость и оптимизацию передачи ресурсов. С появлением MPLS-сетей происходит их изучение с точки зрения влияния самоподобных свойств трафика на качество обслуживания сети. В работах [3,4] исследовано влияние степени самоподобия трафика на сходимость реальной сети при управлении трафиком с различным качеством обслуживания QoS (Quality of Service). В [5, 6] рассмотрены зависимости характеристик QoS, такие как задержки, джиттер и потери пакетов от параметров самоподобия входящего трафика. Показано, что основное влияние на ухудшение характеристик QoS оказывают большие значения показателя Херста, как степени самоподобия, и коэффициента дисперсии, определяющего уровень всплесков трафика.

В настоящее время рассматривается несколько подходов, направленных на уменьшение влияния самоподобности потоков в MPLS-сетях. В работе [7] рассмотрены методы управления трафиком в сетях MPLS с динамически изменяющимися весами путей, зависящими от степени самоподобия входного потока; представлены результаты имитационного моделирования в малых сетях. При другом подходе, представленном в [8], предложен метод маркировки потоков в FEC, учитывающий значения показателя Херста и коэффициента дисперсии. Определение уровня всплесков также может помочь при маркировке потоков трафика, например, в случае, когда более приоритетный трафик в течение долгого времени препятствует прохождению потоков с более низким приоритетом. В работе [9] предлагается подход к сокращению количества LSP (т.е. увеличение количества свободной пропускной способности) и общей стоимости сети; при определении пути для передачи трафика и необходимой пропускной способности учитываются параметры самоподобного трафика.

Целью данной работы является улучшение качества обслуживания в сети MPLS путем предупреждения ее перегрузки при пиковых выбросах трафика, обладающего фрактальными свойствами. Представлено математическое описание функционирования сети MPLS, свойства самоподобного и мультифрактального трафика и разработан алгоритм расчета стоимости маршрутизации с учетом фрактальных свойств трафика.

1. Функционирование сети MPLS [10–12]

В основе архитектуры MPLS лежит принцип обмена меток. Маршрутизаторы коммутации меток LSR (Label Switching Router) помечают пакеты специальными метками, определяющими дальнейший маршрут следования пакета к месту назначения, указывая в метке каждого пакета место назначения, информацию QoS, путь следования пакета до места назначения и др. Все функции классификации пакетов по классам FEC (Forwarding Equivalence Class) на основании необходимого качества обслуживания QoS, а также реализацию дополнительных сервисов фильтрации, явной маршрутизации, выравнивание нагрузки на сеть и управление трафиком берут на себя граничные маршрутизаторы LER (Label Edge Router). Таким образом, весь объем интенсивных вычислений приходится на граничную область, а высокопроизводительная коммутация выполняется в ядре, что позволяет эффективно оптимизировать конфигурацию устройств MPLS в зависимости от их расположения в сети.

Процесс распределения меток между маршрутизаторами приводит к установлению внутри сети MPLS путей с коммутацией по меткам LSP (Label Switching Path). Каждый маршрутизатор LSR содержит таблицу, в соответствии с которой при обработке пакета старое значение метки заменяется новым, после чего пакет отправляется к следующему устройству на пути LSP. Основная проблема заключается в выборе оптимального пути LSP для канала трафика с требованиями QoS в сети. Важным является факт множественности маршрутов: для маршрута между двумя граничными маршрутизаторами может существовать несколько альтернативных путей.

В математической модели сеть MPLS представляется в виде графа $G=(V,E)$, где $V=\{1,2,...,N\}$ и $E=\{1,2,...,M\}$ – набор маршрутизаторов и линий связи между ними соответственно. Все множество узлов в сети MPLS разбивается на два подмножества: $V^+=\{V_i^+, i=1, n_{LER}\}$ – множество граничных маршрутизаторов LER и $V^-=\{V_i^-, i=1, n_{LSR}\}$ – множество коммутаторов меток LSR ($V^+ \cup V^- = V$, $V^+ \cap V^- = 0$).

Предположим, что в каждый момент t на один из маршрутизаторов поступает трафик интенсивностью $\lambda(t)$, относящийся к одному из 8 классов обслуживания $q \in Q$. Каждому классу обслуживания соответствуют значения максимально допустимой задержки τ_q и максимально допустимого процента потерь l_q . Обозначим множество всех каналов QoS трафика Y . В одной линии связи m может существовать множество каналов $y \in Y$. Каждый канал передачи трафика требует полосу пропускания d_y . Требования к пропускной способности могут быть вычислены из договора с клиентом и/или из статистических данных, собранных между входным и выходным маршрутизаторами. Производительность маршрутизации трафика QoS сильно зависит от джиттера, задержки и надежности. Поскольку трафик проходит через LSP, время ожидания на каждом переходе будет иметь негативное влияние на эффективность маршрутизации по джиттеру и задержке. Кроме того, с применением меньшего количества хопов увеличивается надежность передачи трафика по каналам, так как вероятность выхода из строя на LSP уменьшается. Таким образом, у каналов передачи трафика $y \in Y$ с требованиями QoS q есть ограничения на количество хопов в LSP. Для реализации этого ограничения для каждого канала трафика устанавливается допустимое множество путей $P_y = \{p_y^1, ..., p_y^{L_y}\}$ для пути L_y .

Через $\lambda_{p_y}^q(t)$ обозначим, что в момент t на граничный маршрутизатор поступает трафик интенсивностью λ , относящийся к q -му классу обслуживания, который необходимо доставить на выходной маршрутизатор по любым путям из множества P_y , не превышая заданных максимально допустимых значений задержки τ_q и максимально допустимого процента потерь l_q .

Каждый узел сети V в момент времени t характеризуется производительностью μ , коэффициентом потерь $X_V^q(t) \in X$ и средним временем ожидания пакета в очереди $T_V^q(t)$. Переменная $X_V^q(t) \in X$, $V = \overline{1, P_y}$, равна потерям (%) на узле трафика с классом обслуживания q , маршрутизируемого по пути P_y между граничными маршрутизаторами в момент t . Предполагается, что вероятностью искажения пакета в тракте можно пренебречь, и потери происходят исключительно на узлах сети из-за переполнения запоминающего устройства.

Суммарные потери для трафика $\lambda_{p_y}^q(t)$, маршрутизируемого в момент t , не должны превышать максимально допустимого значения для данного класса обслуживания l_q . Потери определяются как отношение количества отброшенных данных к количеству поступивших на обслуживание. Значение величины потерь $X_V^q(t)$ подлежит минимизации.

Ограничения, накладываемые на время задержки, аналогичны: требование $\sum_{V=1}^{P_y} T_V^q(t) \leq \tau_q$ выполняется для всех узлов сети, где $T_V^q(t)$ – среднее время ожидания пакета класса обслуживания q в очереди на узле V , следующего по пути P_y . Выполнение данного ограничения способствует тому, что время доставки пакетов не превысит максимально допустимого значения для заданного класса обслуживания τ_q .

2. Свойства самоподобного и мультифрактального трафика

Известно, что трафик в мультимедийных сетях обладает свойствами самоподобия. Стохастический процесс $X(t)$ является самоподобным с параметром H , если процесс $a^{-H}X(at)$ описывается теми же конечномерными законами распределений, что и $X(t)$. Параметр H , $0 < H < 1$, называемый показателем Херста, представляет собой степень самоподобия процесса. Наряду с этим свойством, показатель $H > 0.5$ характеризует меру долгосрочной зависимости процесса, т.е. убывание автокорреляционной функции $r(k)$ по степенному закону: $r(k) \sim k^{-\beta}$, $0 < \beta < 1$, $H = 1 - (\beta / 2)$.

Если стохастический процесс $X(t)$ обладает однородными фрактальными свойствами, которые определяются одним скейлинговым показателем – параметром Херста H , то в этом случае начальные моменты процесса можно описать формулой $M[|X(t)|^q] = C(q)t^{Hq}$, где $C(q)$ является некоторой детерминированной величиной. Многие фрактальные процессы имеют сложную неоднородную структуру, которая проявляется в изменении статистических свойств реализаций при изменении масштаба агрегирования. В отличие от монофрактальных процессов, где все моменты $M[|X(t)|^q]$ показывают одинаковый скейлинг, мультифрактальные процессы проявляют более сложный закон масштабного поведения $M[|X(t)|^q] = c(q) \cdot t^{qh(q)}$, где $c(q)$ – некоторая детерминированная функция; $h(q)$ – обобщенный показатель Херста, являющийся нелинейной функцией, для которой значение $h(q)$ при $q = 2$ совпадает со значением параметра H [2].

Одной из причин возникновения перегрузок в телекоммуникационных сетях является наличие долгосрочной зависимости (персистентности): чем ближе параметр H к 1, тем медленнее убывают корреляции между данными: $r(k) \sim \frac{1}{k^{2(1-H)}}$. В частности, это означает,

что за высокими значениями трафика данных с большой вероятностью также будут следовать высокие, что не дает достаточно быстро освободиться буферу. Одним из важнейших свойств сетевого трафика, как случайного процесса, является наличие тяжелых хвостов его одномерных функций распределений. Основное свойство случайной величины с тяжелым хвостом состоит в том, что она проявляет высокую изменчивость. Иными словами, выборка из такой случайной величины представляет собой большей частью относительно небольшие значения, однако также содержит и достаточное количество очень больших значений. В трафике это проявляется величиной и количеством всплесков при небольшой интенсивности.

В работах [13-15] показано, что размер очередей в буфере и величина потерь на участке сети, при прохождении самоподобного трафика, определяются показателем Херста и величиной индекса дисперсии $F[\xi] = D[\xi] / M[\xi]$, большие значения которого соответствуют большому разбросу значений потока данных. В [16,17] показано, что очереди и потери, порождаемые трафиком с мультифрактальными свойствами, определяются показателем Херста и степенью нелинейности скейлинга, которая соответствует степени неоднородности трафика, т.е. большому значению разброса данных. Таким образом, основными характеристиками потока, определяющими возникновение перегрузки сети, можно считать показатель Херста и величину нормированного разброса значений трафика $S_{\text{norm}}(\tau) = S / \bar{X}$, где τ – временной интервал, на котором фиксируется проходящий трафик, S – среднее квадратическое отклонение, $\bar{X}$ – интенсивность трафика. В работах [1,2,14] показано, что при значениях $H \geq 0,9$ или при персистентном трафике с нормированным разбросом $S_{\text{norm}} \geq 3$ величина потерь данных превышает 5-10%.

3. Расчет стоимости маршрутизации с учетом фрактальных свойств трафика

В архитектуре MPLS возможен выбор маршрутов на основе отдельных потоков, причем различные потоки, связывающие одну и ту же пару конечных точек, могут следовать по разным маршрутам. Кроме того, при возникновении перегрузки проложенные архитектурой MPLS маршруты могут быть изменены. Протокол маршрутизации, основываясь на данных о состоянии каналов, вычисляет кратчайшие пути (маршруты наименьшей стоимости) между входным граничным маршрутизатором и всеми остальными. В данной работе предложен метод расчета стоимости маршрутизации с учетом самоподобной структуры трафика, что позволяет предупреждать перегрузку сети при пиковых выбросах трафика.

Значение стоимости маршрутизации c_m присваивается линии связи m и может зависеть от ряда параметров, а именно: от скорости, длины и надежности. Стоимость пути p_y^l обозначается C_y^l и равна сумме стоимости линий связи: $C_y^l = \sum_{m \in p_y^l} c_m$. Если $x_y^l(t)$ представляет пропускную способность, которая направляется на допустимый путь p_y^l канала передачи y трафика $\lambda_{p_y^l}^q(t)$, тогда выполняется:

$$\sum_{t \in T; l=1}^{L_y} x_y^l(t) = d_y, \quad \forall y \in Y, \quad \forall l \in \{1, \dots, L_y\}.$$

Целевая функция, минимизирующая стоимость маршрута на множестве путей $P_y = \{p_y^1, \dots, p_y^{L_y}\}$, выглядит следующим образом:

$$\sum_{y \in Y} \sum_{l=1}^{L_y} C_y^l x_y^l(t) \rightarrow \min. \quad (1)$$

При прохождении по сети трафика с сильными фрактальными свойствами требуется своевременное увеличение пропускных способностей линий связи. Чтобы отразить изменение самоподобных свойств потоков, стоимости путей C_y^l обновляются в регулярные промежутки времени и пересчитываются по формуле

$$C_{new}^l = \begin{cases} C_y^l, & H \leq 0,5; \\ C_y^l + (H - 0,5)C_0, & 0,5 < H < 0,9, S_{norm} \leq 1; \\ C_y^l + (H - 0,5)(S_{norm} - 1)C_0, & 0,5 < H < 0,9, 1 < S_{norm} < 3; \\ C_y^l + C_0, & H \geq 0,9 \text{ или } H > 0,5, S_{norm} \geq 3. \end{cases}$$

где $C_y^l = \sum_{m \in p_y^l} c_m$ определяется в соответствии с целевой функцией (1), значение C_0 выбирается администратором сети с учетом ее топологии. Алгоритм маршрутизации не изменяется (стоимость пути $C_{new}^l = C_y^l$), если трафик является обычным пуассоновским потоком ($H = 0,5$) или обладает антиперсистентными свойствами ($H < 0,5$). При $0,5 < H < 0,9$ и малом разбросе данных ($S_{norm} \leq 1$) значение C_y^l увеличивается пропорционально значению показателя Херста ($H < 0,5$). При показателе Херста $0,5 < H < 0,9$ и с большим разбросом данных ($1 < S_{norm} < 3$) значение C_y^l увеличивается пропорционально обоим характеристикам. Стоимость с максимальным значением $C_y^l + C_0$ получается при $H \geq 0,9$ или при персистентном трафике с нормированным отклонением $S_{norm} \geq 3$.

Оценивание параметра Херста осуществляется с помощью дискретного вейвлет-преобразования, что позволяет работать в реальном времени [18]. После перерасчета стоимости всех путей объявление о состоянии путей рассылается между маршрутизаторами.

Выводы

Представлен алгоритм расчета стоимости маршрутизации сети MPLS с учетом фрактальных свойств трафика, который позволяет предупреждать перегрузку сети при пиковых выбросах. *Научная новизна работы* заключается в том, данный метод использует значения показателя Херста и величину нормированного разброса значений трафика, что позволяет применить его к самоподобным и мультифрактальным потокам данных. *Прак-*

тическая значимость заключается в том, что использование предложенного метода позволяет минимизировать стоимость маршрутизации с учетом фрактальных свойств трафика, выбора пути его передачи и требований качества обслуживания. Разработанный метод можно использовать для оптимизации сетевых ресурсов с помощью имитационного моделирования функционирования сети.

Список литературы: 1. Столингс В. Современные компьютерные сети / В. Столингс. СПб.: Питер, 2003. 783 с. 2. Шелухин О. И. Самоподобие и фракталы. Телекоммуникационные приложения / О. И. Шелухин, А. В. Осин, С. М. Смольский. М.: Физматлит, 2008. 368 с. 3. Lropez V. A Bayesian decision theory approach for the techno-economic analysis of an all-optical router / V. Lropez, J.A. Hernandez, J. Aracil, J. P. F. Palacios, O. Gonzalez de Dios // Computer Networks: The International Journal of Computer and Telecommunications Networking Vol. 52 Issue 10. Inc. New York, NY, USA. 2008. P.1916-1926. 4. Casellas R. Packet Based Load Sharing Schemes in MPLS networks // R. Casellas, J. L. Rougier, D. Kohan // Proceedings of the 3rd international IFIP/ACM Latin American conference on Networking. Paris. 2002. P.18-28. 5. Carvalho de P. An Experimental Testbed for Evaluation Topics in Converged Networks / De Carvalho, H. Abdalla, A. M. Soares, P. Barreto, P. Tarchetti, R. Lambert, G. Amvamenze // Departamento de Engenharia Elétrica, Universidade de Brasília, Brazil. 2005. P.503-509. 6. Barreto P. A Traffic Characterization Procedure for Multimedia Applications in Converged Networks / P. Barreto, P. deCarvalho, J. A. M. Soares, H. Abdalla // Proceedings of the 13th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems. Washington, DC, USA. 2005. P.153-160. 7. Vargas-Rosales C. Routing with Wavelet-Based Self-Similarity Estimation / C. Vargas-Rosales, L. J. Manzanero // Computaciyn y Sistemas. vol.8, n.2. Monterrey, Mexico. 2004. P. 119-131. 8. De Carvalho P. Analysis of the influence of self-similar traffic in the performance of real time applications / P. de Carvalho, H. Abdalla Jr., A. M. Soares, P. Solns Barreto, P. Tarchetti // Department of Electrical Engineering, University of Brasilia. 2005. P.480-485. 9. Kyu-Seek Sohn. A Distributed LSP Scheme to Reduce Spare Bandwidth Demand in MPLS Networks / Kyu-Seek Sohn, Seung Yeob Nam, Dan Keun Sung // Communications, IEEE, Vol. 54, Issue:7. 2006. P.1277-1288. 10. Лемешко О.В. Багатоканальний електрозв'язок та телекомунікаційні технології: підручник у 2-х частинах. Ч.1 / О.В. Лемешко, В.А. Лошаков, В.В. Поповський та ін. // Харків: ТОВ "Компанія СМІТ". 2010. 470 с. 11. Олвейн В. Структура и реализация современной технологии MPLS / Вивек Олвейн // Cisco Press. 2004. 480 с. 12. Hussain I. Fault-Tolerant IP and MPLS Networks / I. Hussain // Cisco Press. 2005. 336 с. 13. Кириченко Л. О. Исследование влияния самоподобия трафика при проектировании фрагмента сети / Л. О. Кириченко, Т. А. Радивилова // Нові технології. Науковий вісник Кременчуцького університету економіки, інформаційних технологій і управління. 2007. №1-2. С. 124-129. 14. Kirichenko L. Analysis of network performance under selfsimilar system loading by computer simulation / L. Kirichenko, T. Radivilova // Бионика интеллекта. 2008. №1 (68). С. 158-160. 15. Кириченко Л. О. Влияние методов маршрутизации на качество обслуживания в мультисервисных сетях при самоподобной нагрузке / Л. О. Кириченко, Т. А. Радивилова, Э. Кайали // Восточно-Европейский журнал передовых технологий. 2011. 1/2 (49). С. 15-18. 16. Kirichenko L. Modeling telecommunications traffic using the stochastic multifractal cascade process / L. Kirichenko, T. Radivilova, E. Kayali // Information science and computing. 2012. V. 28. P.55-63. 17. Кириченко Л. О. Моделирование телекоммуникационного трафика с использованием стохастических мультифрактальных каскадных процессов / Л. О. Кириченко, К. А. Демерчян, Э. Кайали, А. Ю. Хабачёва // Радіоелектроніка, інформатика, управління. 2012. №1 (26). С.48-53. 18. Abry P. Wavelet analysis of long-range dependent traffic / P. Abry, D. Veitch // IEEE/ACM Transactions Information Theory. 1998. № 1 (44). P. 2-15.

Поступила в редакцию 24.11.2012

Кириченко Людмила Олеговна, канд. техн. наук, доцент кафедры прикладной математики ХНУРЭ. Научные интересы: самоподобные и мультифрактальные случайные процессы, фрактальный и мультифрактальный анализ временных рядов. Адрес: Украина, 61166, Харьков, просп. Ленина, 14, тел. 70-21-436. E-mail: ludmila.kirichenko@gmail.com.

Радивилова Тамара Анатольевна, канд. техн. наук, доцент кафедры телекоммуникационных систем ХНУРЭ. Научные интересы: телекоммуникационные сети и системы, самоподобные и мультифрактальные информационные потоки. Адрес: Украина, 61166, Харьков, просп. Ленина, 14, тел. 70-21-640. E-mail: tomachka_7@yahoo.com.

Кайали Эйаз, аспирант кафедры прикладной математики ХНУРЭ. Научные интересы: телекоммуникационные сети и системы. Адрес: Украина, 61166, Харьков, просп. Ленина, 14, тел. 70-21-640. E-mail: ludmila.kirichenko@gmail.com.

УСОВЕРШЕНСТВОВАНИЕ МЕТОДА ИЗМЕРЕНИЯ ОСТАТОЧНЫХ НАПРЯЖЕНИЙ В ПОДЛОЖКАХ АРСЕНИД ГАЛЛИЯ

Описывается усовершенствованный метод измерения остаточных напряжений в подложках арсенида галлия путем повышения чувствительности приемного тракта с применением фазочувствительного контура. Разрабатывается структурная схема и алгоритм работы автоматизированного комплекса «Полярон-4». Выполняется метрологическая аттестация разработанного комплекса. Измеряются остаточные напряжения в подложке арсенида галлия диаметром 102 мм, максимальные значения которых составили 8 мПа по краям подложки, вырезанной из верхней части слитка.

1. Введение

Арсенид галлия (GaAs) на сегодняшний день является одним из перспективных полупроводниковых материалов, о чем свидетельствует увеличивающийся рост объемов выпуска слитков GaAs. Основными областями применения GaAs приборов являются быстродействующие микросхемы, СВЧ устройства, мобильные телефоны, планшетные компьютеры. Снижение стоимости изделий, в основе которых лежат GaAs приборы, возможно при снижении затрат на выращивание слитков. Одним из путей решения этой задачи является увеличение диаметра слитков. В свою очередь, увеличение диаметра выращиваемых слитков приводит к увеличению плотности дислокаций и уровня остаточных напряжений. Для измерения остаточных напряжений в подложках арсенида галлия широко используются методы поляриметрии, в основе которых лежит измерение двойного лучепреломления (двулучепреломления) [1].

В том случае, если двулучепреломление является маленьким, необходимо подготавливать образцы толщиной в несколько миллиметров, чтобы увеличить сдвиг фазы, которая пропорциональна толщине образца. Подложки арсенида галлия для промышленного применения изготавливаются толщиной 400 мкм для диаметра подложки 50,8 мм, 600 мкм для подложек диаметром 76,2, и 700 мкм для подложек диаметром 101,6 мм.

Измерение уровня остаточных напряжений для таких толщин требует усовершенствование метода измерения, модернизации аппаратуры в целях увеличения чувствительности измерительного тракта, разработки нового программного обеспечения, включающего в себя цифровую обработку.

Усовершенствование метода измерения уровня остаточных напряжений в подложках GaAs является актуальным как с научной, так и с практической точки зрения. Информация об уровне остаточных напряжений в подложках позволит корректировать технологические процессы выращивания и отжига слитков, а также уменьшить процент брака подложек, вызванного растрескиванием в процессе эпитаксии.

Целью работы является усовершенствование метода измерения остаточных напряжений в подложках GaAs путем увеличения чувствительности прибора и разработки высокочувствительного помехозащищенного приемного тракта.

Для достижения цели необходимо решить следующие задачи:

- анализ метода и аппаратуры поляриметрии в целях определения способов повышения чувствительности приемного тракта;
- разработка структурной схемы усовершенствованного метода;
- метрологическая аттестация автоматического комплекса измерения остаточных напряжений.

2. Постановка задачи

Как правило, в существующих поляриметрах, которые используются для измерения остаточного напряжения в арсениде галлия, применяется эллиптическая поляризация [2,3].

В случае тонкой двулучепреломляющей пластины, которой является подложка GaAs, эллипсоид показателей преломления можно рассматривать как эллипс, общее уравнение которого может быть записано в виде:

$$\frac{y'^2}{n'^2} + \frac{z'^2}{n''^2} = 1, \quad (1)$$

где n' и n'' – неизвестные коэффициенты преломления, направленные вдоль основных осей. Сдвиг фазы, вызванный двулучепреломлением, можно определить по формуле:

$$\delta = \frac{2\pi d}{\lambda} |n' - n''|, \quad (2)$$

здесь d – толщина подложки; λ – длина волны поляризованного света.

Интенсивность входного сигнала на оптическом приемнике в этом случае определяется по формуле [4]:

$$I_t = I_0(\varphi)(1 - R(\varphi))^2 \times (\cos^2(\chi - \varphi) - \sin 2(\chi - \psi) \times \sin 2(\varphi - \psi) \times \sin^2(\delta/2)) \times G(\chi) + I_d(\varphi, \chi), \quad (3)$$

где φ – угол поворота поляризатора; ψ – угол двулучепреломления; χ – угол поворота анализатора; $I_0(\varphi)$ – интенсивность входного излучения, включающая потери излучения в поляризаторе; $R(\varphi)$ – коэффициент отражения подложки GaAs; $G(\varphi)$ – чувствительность приемника излучения, включающая потери в анализаторе; $I_d(\varphi, \chi)$ – темновой ток фотоприемника.

Рассмотрим ограничения метода, которые накладываются на чувствительность приемного тракта поляриметра. Согласно принятому методу измерения [5], I_t необходимо измерять в параллельной плоскости поляризации (случай, когда $\chi = \varphi$) и в перпендикулярной плоскости ($\chi - \varphi = \pi/2$). Для этого выражение (3) может быть переписано в виде:

$$I_t^{\parallel}(\varphi) = I_s(\varphi) \times G^{\parallel}(\varphi)(1 - \sin 2(\varphi - \psi) \times \sin^2(\delta/2)) + I_d^{\parallel}(\varphi) \times G^{\parallel}(\varphi), \quad (4)$$

$$I_t^{\perp}(\varphi) = I_s(\varphi) \times G^{\perp}(\varphi)(\sin 2(\varphi - \psi) \times \sin^2(\delta/2)) + I_d^{\perp}(\varphi) \times G^{\perp}(\varphi), \quad (5)$$

где $I_0(\varphi) = I_0(\varphi)(1 - R(\varphi))^2$.

Тогда общая интенсивность излучения запишется как:

$$I_r(\varphi) = \frac{I_t^{\perp}(\varphi)}{I_t^{\perp}(\varphi) + I_t^{\parallel}(\varphi)} = \frac{G_r(\varphi) \sin^2 2(\varphi - \psi) \sin^2(\delta/2) + I_d^{\perp}(\varphi) G_r(\varphi) / I_s(\varphi)}{(1 + (G_r(\varphi) - 1) \sin^2 2(\varphi - \psi) \sin^2(\delta/2)) + (I_d^{\perp}(\varphi) G_r(\varphi) + I_d^{\parallel}(\varphi)) / I_s(\varphi)}, \quad (6)$$

$G_r(\varphi) = G^{\perp}(\varphi) / G^{\parallel}(\varphi)$ – чувствительность приемного тракта.

Если принять следующие условия:

$$G_r(\varphi) \approx 1 \text{ и } I_d^{\perp}(\varphi), I_d^{\parallel}(\varphi) < I_s(\varphi), \quad (7)$$

тогда (6) можно переписать в следующем виде:

$$I_r(\varphi) \approx G_r(\varphi) \sin^2 2(\varphi - \psi) \sin^2(\delta/2) + I_d^{\perp}(\varphi) G_r(\varphi) / I_s(\varphi). \quad (8)$$

В этом выражении первое слагаемое определяет полезный сигнал, а второе – помеху.

Выполнение условия (7) может быть достигнуто применением специальных решений при разработке аппаратной части полярископа. Из выражения (8) можно видеть, что в том случае, если сдвиг фазы δ сравнительно небольшой, то величина помехи может превысить величину полезного сигнала, что является ограничивающим фактором чувствительности приемного тракта. Вторым ограничением является невыполнение условия

$G_r(\varphi) \approx 1$. Если условие $G_r(\varphi) \approx 1$ не выполняется, уменьшается точность определения параметров δ и ψ , так как $G_r(\varphi)$ модулируется периодическими компонентами $\sin^2 2(\varphi - \psi) \sin^2 (\delta/2)$.

Обеспечить выполнение условия (7) возможно путем использования в схеме поляриметра фазочувствительного контура и высокочувствительного фотоприемника. Увеличить чувствительность приемного тракта возможно путем введения цифровой обработки входного сигнала. В этом случае параметры δ и ψ можно определить как среднее взвешенное для измеренных значений $I_r(\varphi)$, минимизируя при этом влияние шумовой составляющей полезного сигнала. Если производить измерения $I_r(\varphi_j)$ для $(j=0, \dots, J-1)$ в интервале $2\pi/J$ при $0 < \varphi < 2\pi$, то можно записать следующие преобразования синусоидальной и косинусоидальной составляющей входного сигнала:

$$I_{\sin} = \frac{1}{J} \sum_{j=0}^{J-1} I_r(\varphi_j) \sin 4\varphi_j = \frac{1}{4} \sin 4\psi \sin^2 \frac{\delta}{2}, \quad (9)$$

$$I_{\cos} = \frac{1}{J} \sum_{j=0}^{J-1} I_r(\varphi_j) \cos 4\varphi_j = \frac{1}{4} \cos 4\psi \sin^2 \frac{\delta}{2}. \quad (10)$$

Тогда сдвиг фазы δ и угол двулучепреломления ψ может быть определен следующими выражениями:

$$\delta = 2 \arcsin(4(I_{\sin}^2 + I_{\cos}^2))^{1/4}, \quad (11)$$

$$\psi = 4 \arctan \frac{I_{\sin}}{I_{\cos}}. \quad (12)$$

Остаточные напряжения в соответствии с [6] могут быть найдены как:

$$|\sigma| = \frac{\lambda \delta}{\pi d n_0^3} \left| \frac{\sin 2\psi}{p_{44}} \right|, \quad (13)$$

где d – толщина подложки GaAs; n_0 – коэффициент преломления GaAs при отсутствии остаточных напряжений, равный 3,4 [7]; p_{44} – фотоупругая константа, равная 0,0686 [7]; $\lambda = 3,3$ мкм.

3. Усовершенствование метода измерения остаточных напряжений.

Как показано выше, метод измерения остаточных напряжений в подложках GaAs должен реализовать высокую чувствительность приемного тракта. Это возможно при совместном использовании аппаратных и программных решений. Аппаратное решение предполагает введение в схему высокоточного модулятора с фазочувствительным детектором и применение приемника ИК излучения, обладающего низким уровнем темнового тока в диапазоне длин волн от 1 до 4 мкм. Программная часть должна обеспечивать прием, обработку и фильтрацию сигналов δ и ψ .

Для реализации усовершенствованного метода авторами был разработан автоматический комплекс измерения остаточных напряжений «Полярон – 4», структурная схема которого приведена на рис. 1.

Структурная схема включает в себя инфракрасный газовый одномодовый лазер ЛГ-79-1 с, длиной волны 3,39 мкм, мощностью излучения 12 мВт, диаметром пучка 2 мм, вращающийся поляризатор (1) и анализатор (10), которые управляются синхронизаторами вращения СВ (2,11) с точностью определения угла поворота $0,05^\circ$ и частотой вращения 106/с. После четвертьволновой пластины (3) поляризация становится эллиптической и поступает на измеряемую подложку GaAs, укрепленную в сканаторе (5). Сканатор осуществляет перемещение подложки по оси x и y для измерения остаточных напряжений по всей ее поверхности. После прохождения подложки с остаточными напряжениями происходит двулучепреломление и излучение разделяется на нормальный луч и повернутый на угол ψ . Для повышения чувствительности приемного тракта в схему введен механический моду-

лятор (6) с частотой модуляции 3 кГц; модулятор оснащен высокостабильным приводом (7) с ПИД регулятором (8). Для определения сдвига фазы δ в схему введен синхронный детектор СД (9). Для уменьшения темнового тока приемник ИК излучения выполнен на базе фоторезистора PbSe (12) с активным элементом 2х2 мм и термостатированием при температуре -25°C (13). Сигнал с приемника через усилитель (14) поступает на полосовой фильтр (16) с частотой 3 кГц и через АЦП (15) поступает в ЭВМ.

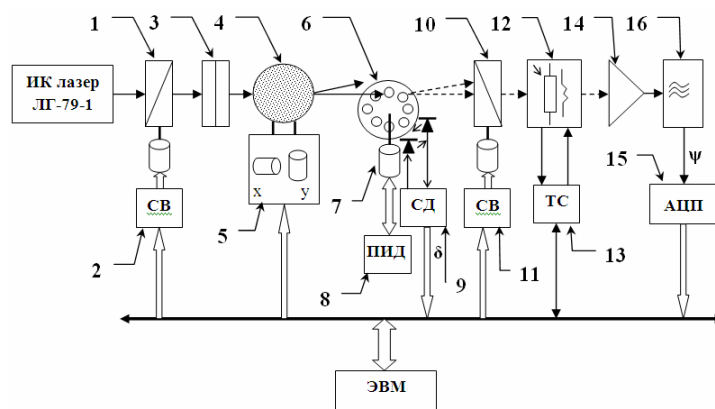


Рис. 1. Структурная схема автоматизированного комплекса измерения остаточных напряжений «Полярон – 4»

ЭВМ формирует сигналы управления синхронизаторами вращения, термостабилизатором и производит вычисления сдвига фазы и угла двулучепреломления по выражениям (9)-(12), а также вычисляет остаточные напряжения (13).

Анализ функциональной схемы автоматизированного комплекса «Полярон-4» позволил синтезировать алгоритм его работы с учетом возможности реализации полной модели сканирования. Данный алгоритм позволяет осуществлять задание шага перемещения исследуемой пластины по оси y , вычислять требуемое перемещение по оси x , производить расчет остаточных напряжений и представлять результаты измерений в удобном для пользователя виде. Разработанный алгоритм приведен на рис. 2.

4. Метрологическая аттестация автоматизированного комплекса «Полярон-4»

Для измерения остаточных напряжений в подложках авторами были проведены метрологические испытания в целях аттестации разработанного автоматизированного комплекса. Для аттестации был использован метод сравнения измеренного значения остаточного напряжения с расчетным значением.

В подложке GaAs были созданы остаточные напряжения по схеме, представленной на рис. 3.

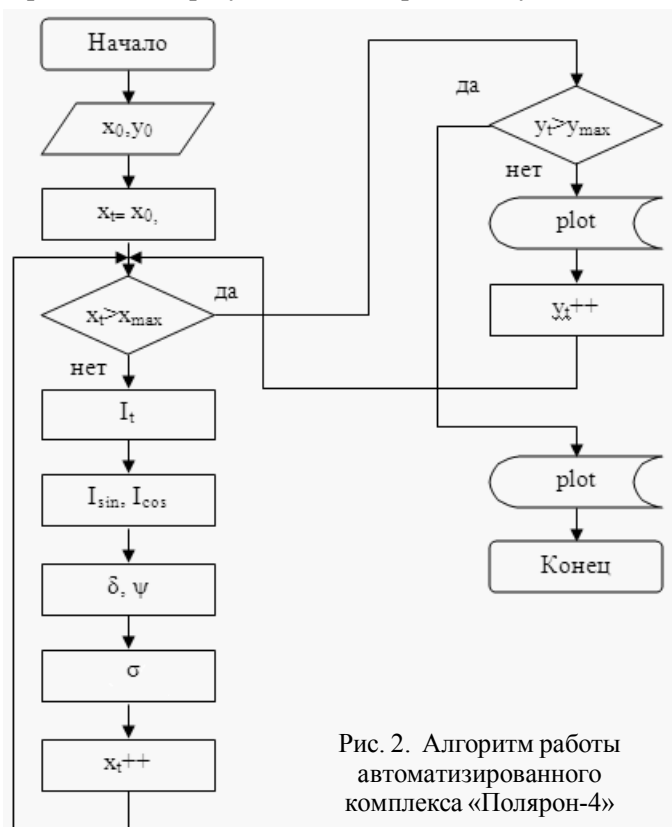


Рис. 2. Алгоритм работы автоматизированного комплекса «Полярон-4»

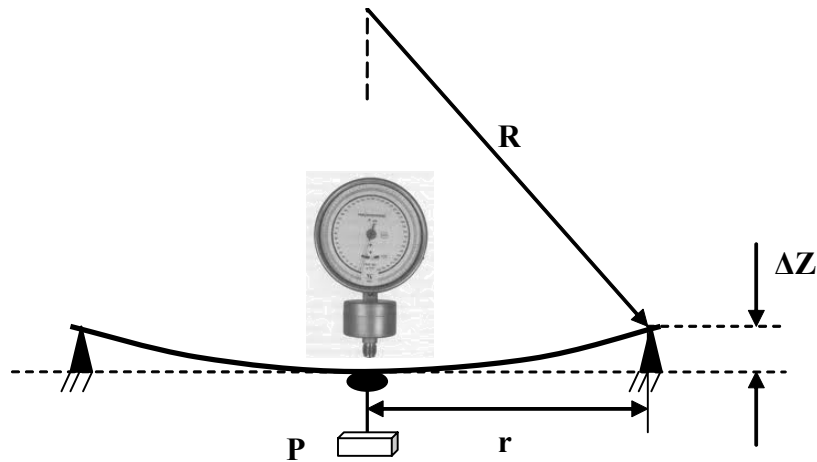


Рис. 3. Схема создания остаточных напряжений в подложках GaAs

В данной схеме подложка диаметром 101,2 мм, толщиной 800 мкм устанавливалась на подставки и создавался ее изгиб грузом P , прикрепленным к центру подложки. Изгиб пластины измерялся индикатором часового типа ИЧ-10. Радиус изгиба подложки определялся по выражению:

$$R = \frac{r}{2 \tan^{-1}(\Delta Z / r)}, \quad (14)$$

где ΔZ – величина изгиба, измеренная индикатором ИЧ-10; r – расстояние от точки подвеса груза до опоры, в нашем случае 37 мм.

Тогда в соответствии с [8] остаточное напряжение определится выражением:

$$\sigma = \frac{E_s h^2}{6(1 - \nu_s) R}, \quad (15)$$

здесь h – толщина подложки (0,6 мм); E_s – модуль Юнга (67,2 ГПа); значение $E_s/(1 - \nu_s)$ в соответствии с [9] было принято равным 123 ГПа.

Для метрологической аттестации нами были измерены 10 подложек при различных величинах изгиба. Рассчитанные значения сравнивались с измеренными значениями остаточного напряжения комплексом «Полярон - 4». С помощью статистических методов нами было получено тарировочное уравнение для вычисления остаточного напряжения, которое записывается в виде:

$$\sigma = 62,0733 - 22,3 \cdot \sigma_{\text{и}} + 2,22222 \cdot \sigma_{\text{и}}^2, \quad (16)$$

где $\sigma_{\text{и}}$ – остаточное напряжение, вычисленное по выражению (13).

Метрологическая аттестация комплекса «Полярон - 4», как экспериментального образца средств измерений, проводилась в соответствии с ГОСТ 8.326-89 «Метрологическая аттестация средств измерений» в центральной заводской лаборатории ООО «Силикон» при участии отдела дифракционных исследований структуры полупроводников Института физики полупроводников имени В. Е. Лашкарева НАН Украины.

Программа метрологической аттестации заключалась в сравнении измеренных значений остаточных напряжений с измерениями остаточных напряжений эталонным прибором. В качестве эталонного прибора использовался высокоразрешающий рентгеновский дифрактометр X'Pert PRO MRD. Измерения производились в фиксированных точках, ориентированных по главным кристаллографическим осям с шагом 20 мм.

Относительная погрешность измерений остаточных напряжений рассчитывалась по формуле:

$$\sigma = \frac{\sigma_s - \sigma_i}{\sigma_i} \cdot 100\%, \quad (17)$$

где σ_s – остаточное напряжение, измеренное рентгеновским дифрактометром; σ_i – остаточное напряжение, измеренное комплексом «Полярон - 4».

По результатам относительная погрешность измерения остаточных напряжений в подложках GaAs составила 10 %.

5. Практические результаты

Используя автоматизированный комплекс, мы провели измерения остаточных напряжений в подложках арсенида галлия, полученных из слитка марки АГЧХ диаметром 4'', выращенного в направлении [100].

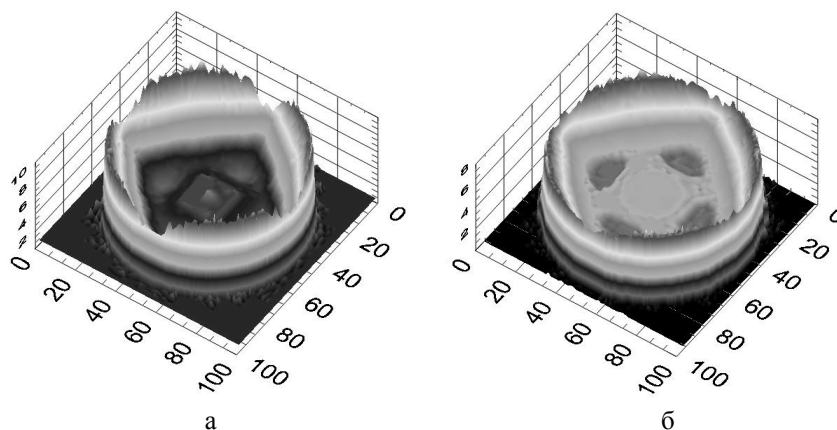


Рис. 4. Результат измерения остаточных напряжений в подложке GaAs: а – подложка из верхней части слитка; б – подложка из средней части слитка

Результаты измерения остаточных напряжений показали, что в подложках, полученных из верхней части слитка, их уровень составляет порядка 8 мПа по краям подложки и снижается до 4 мПа к центру подложки. В подложках вырезанных из средней части слитка уровень остаточных напряжений по краям подложки составляет 6 мПа и снижается к ее центру до 2 мПа (рис.4).

Анализ полученных результатов показал, что остаточные напряжения распределяются по плоскости симметрично, в виде четырехкратной симметрии. Уровень остаточных напряжений в направлении [100] имеет более высокие значения, чем в направлении [010].

6. Выводы

В результате проведенных исследований был усовершенствован метод измерения уровня остаточных напряжений в подложках GaAs. Определены факторы существующего метода, не позволяющие увеличить точность измерения остаточных напряжений в подложках арсенида галлия.

Разработана структурная схема и алгоритм работы автоматизированного комплекса «Полярон-4», позволяющего измерять остаточные напряжения в подложках GaAs толщиной от 400 мкм. Получена тарировочная характеристика комплекса по результатам расчетных значений остаточных напряжений в подложках GaAs.

Выполнена метрологическая аттестация комплекса. Относительная погрешность составила 10 %.

Показано, что подложки, вырезанные из верхней части слитка, имеют более высокий уровень остаточного напряжения, чем подложки, вырезанные из средней части. Остаточные напряжения на краях подложки больше остаточных напряжений в центре примерно на 50%.

Показано, что распределение остаточных напряжений в подложках GaAs имеет четырехкратную симметрию.

Список литературы: 1. Оксанич А.П., Притчин С.Э., Краскевич В.Е., Батареев В.В. Автоматизированный комплекс для измерения внутренних напряжений в пластинах GaAs // Складні системи і процеси. Запоріжжя. 2006. № 2. С. 40-50. 2. M. Yamada, Quantitative photoelastic measurement of residual strains in undoped semi-insulating gallium arsenide, Appl. Phys. Lett. 47.1985. С. 365–367. 3. Ткаченко С.В., Притчин С.Э., Сиора А.С., Дервянко И.А. Автоматизированная система определения внутренних напряжений в

пластинах полупроводников // Вісник КДПУ імені Михайла Остроградського. Вип. 6/2007 (47). Частина 1. С. 33-35. **4.** *Born M., Wolf E.* Principles of Optics // Pergamon, Oxford, 1970. pp. 665-718. **5.** *Lesniak, J.R., Zickel, M.J., Welch, C.S., Johnson, D.F.* An innovative polariscope for photoelastic stress analysis // SEM Spring Conference on Experiments Mechanics, 1997. P. 298-301. **6.** *Nye J.F.* Physical Properties of Crystals. // Oxford University Press, 1972. P. 235-259. **7.** *Adachi S., Oe. J.K.* Internal strain and photoelastic effects in $\text{Ga}_{1-x}\text{Al}_x\text{As}/\text{GaAs}$ and $\text{In}_{1-x}\text{Ga}_x\text{As}_y\text{P}_{1-y}/\text{InP}$ crystals // J. Appl. Phys. 1983. v54. P. 6620 – 6628. **8.** *Timoshenko S., Woinowsky-Krieger S.* Theory of Plates and Shells // McGraw-Hill, Inc., New York, 1959. P. 38. **9.** *Kirkby P.A., Selway P.R., Westbrook L.D.* Photoelastic waveguides and their effect on stripe-geometry GaAs/Ga $_{1-x}\text{Al}_x\text{As}$ lasers // J. Appl. Phys. 1979. v50. P. 4567 – 4580.

Поступила в редколлегию 18.12.2012

Оксанич Анатолий Петрович, д-р техн. наук, профессор, директор НИИ технологии полупроводников и информационно-управляющих систем Кременчугского национального университета имени Михаила Остроградского, зав. кафедрой информационно-управляющих систем. Научные интересы: методы и аппаратура контроля структурно-совершенных полупроводниковых монокристаллов. Адрес: Украина, 39600, Кременчуг, ул. Первомайская, 20, тел.: (05366) 30157. Email: oksanich@kdu.edu.ua.

Притчин Сергей Эмильевич, канд. техн. наук, доцент кафедры информационно-управляющих систем Кременчугского национального университета имени Михаила Остроградского. Научные интересы: автоматизация процессов управления производством полупроводниковых материалов. Адрес: Украина, 39600, Кременчуг, ул. Первомайская, 20, тел.: (05366) 30157. Email: spmywork@ukr.net.

Тербан Виктор Андреевич, канд. техн. наук, главный инженер ЧП «Галар». Научные интересы: оборудование и производство материалов электронной техники. Адрес: Украина, Кировоградская область, Светловодск, ул. Заводская, 3, тел. (5236) 71535. Email: spmywork@ukr.net.