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 159

Харьков
2012

УДК 62-50

В сборнике представлены результаты исследований, касающихся компьютерной инженерии, управления, технической диагностики, автоматизации проектирования, оптимизированного использования компьютерных сетей и создания интеллектуальных экспертных систем. Предложены новые подходы, алгоритмы и их программная реализация в области автоматического управления сложными системами, оригинальные информационные технологии в науке, образовании, медицине.

Для преподавателей университетов, научных работников, специалистов, аспирантов.

У збірнику наведено результати досліджень, що стосуються комп'ютерної інженерії, управління, технічної діагностики, автоматизації проектування, оптимізованого використання комп'ютерних мереж і створення інтелектуальних експертних систем. Запропоновано нові підходи, алгоритми та їх програмна реалізація в області автоматичного управління складними системами, оригінальні інформаційні технології в науці, освіті, медицині.

Для викладачів університетів, науковців, фахівців, аспірантів.

Свидетельство о государственной регистрации
печатного средства массовой информации КВ № 12073-944ПР от 07.12.2006 г.

Адрес редакционной коллегии: Украина, 61166, Харьков, просп. Ленина, 14, Харьковский национальный университет радиоэлектроники, комн. 321, тел. 70-21-326

СОДЕРЖАНИЕ

УДК 621.3.06

И.В. ЛИСИЦКАЯ, А.А. НАСТЕНКО, К.Е. ЛИСИЦКИЙ

БОЛЬШИЕ ШИФРЫ – СЛУЧАЙНЫЕ ПОДСТАНОВКИ. СРАВНЕНИЕ ДИФФЕРЕНЦИАЛЬНЫХ И ЛИНЕЙНЫХ СВОЙСТВ ШИФРОВ, ПРЕДСТАВЛЕННЫХ НА УКРАИНСКИЙ КОНКУРС, И ИХ УМЕНЬШЕННЫХ МОДЕЛЕЙ

Дополнительно обосновывается справедливость гипотезы о том, что большие шифры асимптотически являются случайными подстановками. Выполняется сравнение дифференциальных и линейных свойств шифров, представленных на украинский конкурс, и их уменьшенных моделей. Рассматриваются линейные и дифференциальные показатели финалистов конкурса AES шифров Rijndael, Serpent, а также Threefish. Устанавливается, что по дифференциальным и линейным показателям украинские шифры Калина, Мухомор и Лабиринт превосходят признанного мирового лидера блочного симметричного шифрования.

Введение

Одним из актуальнейших направлений развития современной криптологии считается совершенствование методологии оценки показателей стойкости блочных симметричных шифров к атакам линейного и дифференциального криптоанализа. В этой работе мы продолжаем обоснование новой точки зрения (новой идеологии) в вопросах оценки стойкости блочных симметричных шифров к атакам дифференциального и линейного криптоанализа [1], которая строится на установленном в ходе исследований факте, что все современные блочные шифры после нескольких начальных циклов шифрования приобретают свойства случайных подстановок соответствующей степени [2-5 и др.].

Основой развиваемого подхода является положение, в соответствии с которым большие шифры повторяют свойства своих уменьшенных моделей. В частности, в наших работах [6,7] было показано, что большие версии шифров при использовании их в режиме зашифрования укороченных (16-битных и 32-битных) блоков данных повторяют законы распределения вероятностей переходов XOR таблиц и таблиц смещений линейных аппроксимаций, свойственные соответствующим законам распределения вероятностей своих уменьшенных версий. Последние, в свою очередь, после нескольких начальных циклов зашифрования приходят к законам распределения вероятностей переходов XOR таблиц и смещений таблиц аппроксимаций случайных подстановок. Но выполненные эксперименты коснулись только шифров Rijndael, ГОСТ 28147-89 и FOX.

Продолжая развивать это направление, в настоящей работе мы представляем материалы по дополнительному обоснованию справедливости отмеченной выше гипотезы. Теперь сравниваются дифференциальные и линейные свойства шифров, представленных на украинский конкурс: Калина, Мухомор, Лабиринт и их уменьшенных моделей. Мы дополнили этот список ещё тремя шифрами: Rijndael, Serpent и Threefish.

1. Методика выполнения исследований

Здесь мы воспользовались методикой исследований, предложенной в работе [6]. Большой шифр применяется как малый для шифрования блоков данных уменьшенной длины (зашифрованные блоки данных тоже усекаются до необходимого размера), при этом сохраняются все преобразования и внутренние связи большого шифра. Самое же примечательное при таком подходе – это то, что появляется возможность применить весь наработанный аппарат изучения показателей случайности малых версий шифров для определения показателей случайности больших шифров.

Правомерность использования такого подхода оправдывается тем, что все украинские шифры построены на основе сбалансированных схем и байтовых 8x8 подстановок. Поэтому они не имеют особых дифференциальных характеристик, как в шифре DES, получающихся из-за наличия циклов с переходами обнуляющего типа [8].

В первом случае можно использовать 16-битные блоки открытых и соответствующих им зашифрованных текстов. Здесь мы приходим к условиям, в которых исследовались малые модели шифров [2-5 и др.]. Вычислительных ресурсов хватает для построения всей дифференциальной таблицы (большой шифр работает как его уменьшенная 16-битная версия).

Очевидно также, что при рассматриваемом подходе можно построить и уменьшенную таблицу смещений линейных аппроксимаций.

Во втором случае можно строить переходы для отдельной строки дифференциальной таблицы или смещения таблицы линейных аппроксимаций при использовании большого шифра для шифрования 32-битных блоков данных. Мы в этой работе воспользуемся 16-битными фрагментами входныъх и выходных блоков данных.

Дальнейший материал посвящен изложению результатов применения этой методики для исследования дифференциальных и линейных свойств украинских шифров, а также шифров Rijndael (AES), Serpent, Threefish и IDEA.

2. Сравнение дифференциальных и линейных показателей больших шифров и их уменьшенных моделей

*Дифференциальные свойства блочных симметричных шифров.*В первой серии экспериментов были рассмотрены дифференциальные свойства шифров, представленных на украинский конкурс, и их малых версий. В табл. 1 обобщены результаты таких экспериментов для шифров Мухомор, Калина, Лабиринт и ADE для 30 случайно взятых мастер-ключей. Описание малых версий этих шифров можно найти в работе [8].

Таблица 1

Поцикловые значения максимумов переходов XOR таблиц для полных версий украинских шифров

Число циклов	Калина	ADE	Лабиринт	Мухомор
1	19,47	65536	18	19,13
2	19,0	20	20	18,8
3	19,13	20	18	19,4
4	19,2	18	20	19,13
5	19.27	18	20	19,07
6	18,87	20	20	19,6
7	19,47	20	20	19,27
8	19.2	18	18	19,13
9	19,0	18	18	19,13
10	19,33	18	18	19,276

Заметим, что рассмотренные шифры по спецификациям имеют Мухомор-128 - 11 циклов зашифрования, Лабиринт - 8 циклов, Калина 128/256 - 14 циклов, ADE 128/128 - 10 циклов, Rijndael - от 10-ти до 14-ти циклов в зависимости от длины блока и ключа. В табл. все эксперименты приведены к 10-ти циклам зашифрования.

В табл.2 представлены результаты экспериментов с малыми версиями этих же шифров (для 30 ключей зашифрования).

Таблица 2

Поцикловые значения максимумов переходов таблиц полных дифференциалов для уменьшенных моделей украинских шифров

Число циклов	Мини-Калина	Мини-ADE	Мини-Лабиринт	Мини-Мухомор
1	3732,48	16384	37,5	65536
2	382,4	3353,6	**19,04**	5770,24
3	**19,36**	307,2	19,24	1802,24
4	19,14	20,54	19,04	125,53
5	19,2	**19,08**	19,14	29,7
6	19,36	19,24	19,24	**18,88**
7	18,93	18,87	19,33	18,67
8	19,27	19,27	18,67	19,00
9	18,93	19,20	19,00	18,00
10	18,87	18,73	18,00	18,67

Из представленных данных видно, что большие шифры, также как и малые их модели, действительно после нескольких (а то и сразу) начальных циклов зашифрования приходят к стационарным состояниям, повторяющим дифференциальные показатели случайных подстановок.

Примечательно, что малые модели шифров во всех случаях показывают динамику перехода к стационарному состоянию худшую, чем их большие прототипы. Наибольшее различие по динамике перехода (пять циклов) имеет шифр Мухомор. Это означает, что в его уменьшенной модели не удалось повторить все особенности оригинального преобразования. Мы уже отмечали в работе [9], что в этом шифре не удалось отмасштабировать SL-преобразование, и оно было заменено случайной подстановкой. Для шифра ADE разница для большой и уменьшенной модели составила четыре цикла, для Калины - три, для Лабиринта - два. Нужно также не забывать, что большие шифры имеют существенно увеличенный размер битового входа в шифры, поэтому они быстрее становятся случайными подстановками.

В табл. 3 представлены результаты оценки поцикловых значений максимумов таблиц линейных аппроксимаций украинских мини-шифров вместе со значениями среднеквадратических отклонений. И в этом случае эксперимент для каждого шифра проводился на 30 ключах зашифрования.

Таблица 3

Поцикловые значения максимумов смещений таблиц линейных аппроксимаций мини-шифров со значениями среднеквадратических отклонений (30 ключей)

Число циклов	Мини-Калина	Мини-Мухомор	Мини-Лабиринт	Mini-ADE
1	9671,1±867	32768±0	3178±777	16384
2	3370,6±301	12839,3±1031	980±193	9093,10±94,37
3	836,8±15	6400±697	825,4±14	3509,8±62,37
4	832,2±21	1797,6±347	825,6±23	828,56±7,58
5	838,6±21	837,8±47	817,2±11	820,52±5,48
6	835,5±33	815,6±24	824±21	819,92±5,81
7	821,5±22	817,2±20	823,4±30	818,55±5,35
8	827,3±18	815,8±15	833,6±35	837,34±5,91
9	813,3±21	815,5±15	824,8±24	814,95±6,21
10	834±28	810±17	819±17	822,54±7,13

Результаты вычислительного эксперимента по определению поцикловых средних значений максимумов линейных корпусов для больших версий украинских шифров представлены в табл. 4 в виде модульных значений максимальных смещений.

Таблица 4

Поцикловые значения максимумов смещений таблиц линейных аппроксимаций шифров со значениями среднеквадратических отклонений (30 ключей)

Число циклов	Калина 30 ключей	Мухомор 30 ключей	Лабиринт 1 ключ	ADE 5 ключей
1	11008,392± 1785,34	824,742± 20,1286	– 790	32768
2	817,271± 27,6348	818,621± 25,9742	839	3914.,4
3	817,718± 21,3851	827,431± 21,2352	-816	9523,2
4	814,19± 26,7792	824,193± 17,8115	832	1224,6
5	837,349± 28,2712	831,753± 25,7731	885	811,2
6	810,733± 29,3801	814,155± 28,9121	810	832,8
7	820,384± 20,752	820,975± 20,2673	– 834	827,4
8	837,917± 23,2539	823,024± 18,853	835	822,4
9	809,273± 22,186	810,196± 22,9352	– 809	826,8
10	821,755± 25,5737	821,316± 25,849	– 806	802,8

Как видно из представленных результатов, шифр Калина обладает средним значением максимума таблицы линейных аппроксимаций (820), практически не зависящим от используемых ключей шифрования (среднеквадратическое отклонение не превышает 30), харак-

терным для случайной подстановки степени 2^{16}. Это значение БСШ Калина достигает после 2-х циклов шифрования, что примечательно, так как предельных дифференциальных показателей случайной подстановки данный шифр достигает после 3-х циклов шифрования. Из этого факта можно сделать вывод, что эффективность цикловых преобразований в отношении защищенности от атак линейного криптоанализа у БСШ Калина немного выше, чем в отношении защищенности от атак дифференциального криптоанализа.

БСШ Мухомор достигает среднего значения максимума смещения линейного корпуса (820), характерного для случайной подстановки степени 2^{16}, уже после первого цикла шифрования, так же, как и среднего значения максимума дифференциального перехода (19), характерного для случайной подстановки аналогичной степени. Как видно, шифр Мухомор превосходит по эффективности цикловых преобразований в отношении защищенности от атак как дифференциального, так и линейного криптоанализа шифр Калина.

Близкие к Мухомору показатели показывает и шифр Лабиринт. Шифр ADE приходит к стационарному значению максимума смещения лишь на пятом цикле (здесь использована исходная версия шифра ADE [10] - без коррекции).

Следует заметить, что в табл. 4 представлены результаты для разного объёма ключевого материала. Дело в том, что базовый алгоритм подсчёта смещений таблиц линейных аппроксимаций вычислительно оказался более сложным по сравнению с алгоритмом построения дифференциальных таблиц. Удалось существенно продвинуться вперед в связи с найденным в Интернете описанием быстрого алгоритма расчёта линейных аппроксимационных таблиц (ЛАТ) [11].

В табл. 5 проведены результаты оценки временных затрат на расчет 1 строки ЛАТ, ЛАТ для всего набора циклов (максимальное число циклов 10), а также ЛАТ для всего набора циклов на 30-ти ключах.

Таблица 5

Временные затраты на построение линейной аппроксимационной таблицы

Алгоритм	$T_{1\ строки}$	$T_{1\ таблицы}$	$T_{10\ раундов}$	$T_{30\ ключей}$
Базовый	2,5 мин (2^{32} операций)	113 дней (2^{48}операций)	3 года ($\approx 2^{51}$операций)	92 года ($\approx 2^{56}$операций)
Ускоренный	$\approx$ 2 mc	2,5 мин (2^{32}операций)	25 мин ($\approx 2^{35}$операций)	12,5 ч ($\approx 2^{40}$операций)

Далее приведем результаты исследования дифференциальных и линейных свойств ещё для трех современных шифров: Rijndael, Serpent, Threefish. Были взяты полные версии этих шифров. Длина ключа и блока для Rijndael-я и Serpent-а одинакова и равна 128 битам, а в реализации шифра Threefish использовалась длина для блока и ключа 512 бит.

В табл. 6 представлены результаты распределения значений ячеек таблицы XOR-разностей для 16-битных сегментов шифртекстов всех трех шифров (Rijndael – 10 циклов шифрования, Serpent – 32 цикла, Threefish –72 цикла) и случайной подстановки степени 2^{16}.

Из табл. 6 следует, что распределения значений ячеек таблиц XOR-разностей для 16-битных сегментов шифртекстов для всех трех шифров после всех циклов преобразований очень близки к результатам, полученным вычислительным путём для случайной подстановки.

Результаты свидетельствуют также о том, что результирующие дифференциальные свойства шифров не связаны со свойствами S-блоков шифра, а являются общим свойством шифра, как случайной подстановки. Хорошим примером служат результаты анализа шифра Threefish, в котором не используются S-блоки.

Далее в табл. 7 представлены поцикловые значения максимумов полных дифференциалов для 16-битных сегментов. Для криптоалгоритмов Serpent и Threefish показаны первые 10 циклов, чего вполне достаточно для обозрения того, что шифры реализуют свой асимптотический показатель среднего значения максимума полных дифференциалов. Вычисления проводились с использованием 10 различных ключей для каждого шифра.

Таблица 6

Сравнение распределений значений ячеек таблицы XOR-разностей для 16-битных сегментов шифртекстов БСШ и случайной подстановки порядка 2^{16}

Значение перехода $2k$	Количество переходов (расчет для подстановки)	Количество переходов (Rijndael)	Количество переходов (Serpent)	Количество переходов (Threefish)
0	2605070418	2604948298	2604933270	2604928534
2	1302484861	1302476170	1302501597	1302508996
4	325626184	325620651	325614188	325612232
6	54271858	54268159	54265223,5	54265483,9
8	6784085	6783987,73	6782692,47	6782055,87
10	678418	678135,4	678425,133	678148,067
12	56535	56512,2	56524,067	56449,467
14	4038	4045,33	4027,133	4061,533
16	252	252,6	261,267	249,467
18	14	13,93	15,267	13,667
20	1	0	0	0

Таблица 7

Поцикловые значения максимумов полных дифференциалов для 16-битных сегментов

Число циклов, r	MAX (Rijndael)	MAX (Serpent)	MAX (Threefish)
1	65536±0	18,93	65536
2	3652,26±630,31	19,24	65536
3	19,07± 1,44	18,64	65536
4	19,07±1,00	18,33	42440,04
5	18,87±1,23	18,75	30704,23
6	19,13±0,99	19,21	9534,57
7	19,27± 1,09	18,98	37,75
8	19,13± 1,43	18,37	19,27
9	19,06± 1,23	19,24	18,78
10	19,33± 1,30	19,63	18,44

Приведенные результаты также говорят о том, что шифрующие преобразования асимптотически для различных ключей зашифрования ведут себя как случайная подстановка, т.е. и для них оказываются справедливыми расчетные соотношения, которые используются для случайной подстановки. Представленные для анализа шифры по-разному выходят на асимптотический показатель среднего значения максимума. Rijndael – после 4-го цикла, шифр Serpent выходит на данный показатель уже с 1-го цикла шифрующего преобразования за счет наличия в алгоритме начальной перестановки. Threefish выходит на асимптотический показатель среднего значения максимума только с 8-го цикла. На основе полученных результатов можно, тем не менее, предложить подход к сравнению эффективности решений по построению алгоритмов шифрования (при прочих равных условиях) в виде минимального числа циклов алгоритма, при котором реализуется асимптотический показатель среднего значения максимума полных дифференциалов.

Линейные свойства блочных симметричных шифров. Анализ линейных свойств был выполнен при помощи быстрого алгоритма построения таблиц линейных аппроксимаций [11]. Для всех трех шифров был выполнен подсчет максимумов линейных корпусов с использованием 10 различных случайно сгенерированных ключей.

Для алгоритмов Serpent и Threefish показаны первые 10 циклов, чего вполне достаточно для того, чтобы удостовериться, что шифры приходят к своим асимптотическим значениям максимумов полных дифференциалов (свойственным случайной подстановке степени 2^{16}).

В табл. 8 приведены математические ожидания максимальных значений смещений линейных корпусов для всех исследуемых шифров в зависимости от числа циклов шифрования r.

Таблица 8

Математические ожидания максимальных смещений линейных корпусов полных моделей шифров

Число циклов, *r*	MAX (Rijndael)	MAX (Serpent)	MAX (Threefish)
1	0	810,4	32768
2	9284,27± 657,45	825,0667	32680,93
3	818,47± 26,88	828,2667	31306,13
4	815,0± 28,20	825,9333	23730,93
5	818,5± 18,53	828,4667	19722,67
6	815,97± 20,18	824,8667	19722,67
7	832,1± 33,19	820,3333	7899,8
8	823,13± 23,57	817,5333	844,067
9	829,9± 33,57	820,4	822,13
10	827,4± 25,29	816,6	815,8

Представленные результаты свидетельствуют о том, что и по линейным показателям блочные симметричные шифры после определенного начального числа циклов приходят к показателям случайной подстановки: Rijndael – после 4-го цикла, шифр Serpent приходит к установившемуся значению максимума линейного корпуса, характерному для случайных подстановок, уже с 1-го цикла шифрующего преобразования. Threefish выходит на асимптотический показатель среднего значения максимума только с 8-го цикла.

Таким образом, дифференциальные и линейные свойства шифрующих преобразований исследуемых шифров (при заявленном числе циклов преобразования) являются одним из проявлений свойств случайных подстановок.

Выводы

Рассмотрены дифференциальные и линейные свойства шифров, представленных на украинский конкурс по выбору национального стандарта шифрования, и шифров из числа лидеров конкурса NESSIE.

Результатами приведенных исследований подтверждено одно из центральных положений развиваемого в работе [1] подхода, в соответствии с которым большие шифры повторяют свойства своих уменьшенных моделей. Конечно, здесь речь идёт о приближениях, но для нас важен сам факт прихода больших шифров к стационарному состоянию, свойственному случайной подстановке.

Главный результат наших исследований состоит в том, что получены дополнительные свидетельства того, что и большие шифры асимптотически становятся случайными подстановками. А это означает, что показатели стойкости блочных симметричных шифров могут быть получены расчётным путём из формул, определяющих значения максимумов XOR таблиц и смещений таблиц линейных аппроксимаций, полученных для случайных подстановок [12,13].

Список литературы: 1. *Лисицкая И.В.* Методология оценки стойкости блочных симметричных шифров // АСУ и приборы автоматики. 2011. № 143. С. 123-133. **2.** *Исследование* циклических и дифференциальных свойств уменьшенной модели шифра Лабиринт / В.И. Долгов, И.В. Лисицкая, А.В. Григорьев, А.В. Широков // Прикладная радиоэлектроника. 2009. Т.8, №3. С. 283-289. **3.** *Исследование* дифференциальных свойств мини-шифров Baby-ADE и Baby-AES / В.И. Долгов, А.А. Кузнецов, Р.В. Сергиенко, О.И. Олешко // Прикладная радиоэлектроника. 2009. Т.8, №.3. С. 252-257. **4.** *Долгов В.И.* Дифференциальные свойства блочных симметричных шифров, представленных на украинский конкурс / В.И. Долгов, А.А. Кузнецов, С.А. Исаев // Электронное моделирование. 2011. Т.33, № 6. С. 81-99. **5.** *Кузнецов А.А.* Линейные свойства блочных симметричных шифров, представленных на украинский конкурс / А.А. Кузнецов, И.В. Лисицкая, С.А. Исаев // Прикладная радиоэлектроника. 2011. Т.10, №2. С. 135-140. **6.** *Лисицкая И.В.* Большие шифры - случайные подстановки / И.В. Лисицкая, А.А. Настенко // Радиотехника. 2011. Вып. 166. С. 50-55. **7.** *Лисицкая И.В.* Дифференциальные свойства шифра FOX / И.В. Лисицкая, Д. С. Кайдалов // Прикладная радиоэлектроника. 2011. Т.10, №2. С. 122-126. **8.** *Долгов В.И.* О роли схем разворачивания ключей в атаках на итеративные шифры / В.И. Долгов, А.А. Настенко // Прикладная радиоэлектроника, 2012. № 30. С. 247-252. **9.** Криптографические свойства уменьшенной версии шифра IМухоморI. / И.В. Лисицкая, О.И. Олешко, С.Н. Руденко и др. // Спеціальні телекомунікаційні системи та захист інформації. Збірник наукових праць, Київ. 2010. Вип. 2(18). С. 33-42. **10.**

Кузнецов А.А. Симметричный криптографический алгоритм ADE (Algorithm of Dynamic Encryption). / А.А. Кузнецов, Р.В. Сергиенко, А.А. Наумко // Прикладная радиоэлектроника. Харьков: ХТУРЭ. 2007. Т. 6, №2. С. 241-249. **11.** *Krzysztof Chmiel*. On Differential and Linear Approximation of S-box Functions / Biometrics, Computer Security Systems and Artificial Intelligence Applications. / Edited by Khalid Saeed, Jerzy Pejas and Romuald Mosdorf // Poland, Springer. 2006. P. 111-120. **12.** *Олейников Р.В.* Дифференциальные свойства подстановок / Р.В. Олейников, О.И. Олешко, К.Е. Лисицкий, А.Д. Тевяшев // Прикладная радиоэлектроника. 2010. Т.9, № 3. С. 326-333. **13.** *Долгов В.И.* Свойства таблиц линейных аппроксимаций случайных подстановок / В.И. Долгов, И.В. Лисицкая, О.И. Олешко // Прикладная радиоэлектроника. Харьков: ХНУРЭ. 2010. Т. 9, № 3. С. 334-340.

Поступила в редколлегию 23.06.2012

Лисицкая Ирина Викторовна, д-р техн. наук, доцент кафедры безопасности информационных технологий ХНУРЭ. Научные интересы: криптография, методы криптоанализа. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел.: +38 (057) 702-14-25, E-mail: ai@kture.kharkov.ua.

Настенко Андрей Александрович, аспирант кафедры безопасности информационных технологий ХНУРЭ. Научных интересов: криптография, методы криптоанализа. Адрес: Украина, 61166, Харьков, пр. Ленина, 14.

Лисицкий Константин Евгеньевич, студент ХНУРЭ. Научные интересы: криптография, методы криптоанализа. Адрес: Украина, 61166, Харьков, пр. Ленина, 14.

УДК 658.512.011:681.326:519.713

МУРАД АЛИ АББАС, БАГХДАДИ АММАР АВНИ АББАС, ХАХАНОВ В.И., ЛИТВИНОВА Е.И., ДАХИРИ ФАРИД

ТЕХНОЛОГИИ ВОССТАНОВЛЕНИЯ РАБОТОСПОСОБНОСТИ МУЛЬТИПРОЦЕСОРНЫХ СИСТЕМ НА КРИСТАЛЛАХ

Проводится обзор мультипроцессорных систем на кристаллах (MPSoC) и технологий восстановления их работоспособности. Описываются архитектуры MPSoC, базовая архитектура самовосстановления, метод обнаружения и исправления ошибок в программируемых логических матрицах FPGA.

Цель исследования – аналитический обзор технологий тестирования, ремонта и диагностирования цифровых систем на кристаллах при их проектировании и верификации, ориентированный на существенное увеличение выхода годной продукции и уменьшение времени ее выхода на рынок микроэлектроники.

Для достижения цели необходимо решить задачи, связанные с обзором технологий встроенного ремонта цифровых систем на кристаллах, инфраструктуры сервисного обслуживания и аппаратных решений.

1. Мультипроцессорные системы на кристалле

Создание мультипроцессорных систем-на-кристалле (MPSoC) является важным направлением цифровой электроники [1-6, 18, 22]. Как правило, MPSoC представляет собой систему на кристалле с более чем одним процессором. В современных MPSoC реализованы сложные многопроцессорные модули, которые позволяют решать задачи построения сетей связи (networking), обработки мультимедийной информации и управления в реальном времени (Real Time, RT) при одновременном выполнении ограничений, связанных с потреблением энергии и размерами кристалла.

Мультипроцессор на основе программируемых логических интегральных схем (field-programmable gate array, FPGA) имеет ряд преимуществ перед специализированными интегральными схемами (application-specific integrated circuit, ASIC). А именно: возможности быстрого прототипирования, создания новых архитектур и методов коммуникации. Поэтому мультипроцессоры на основе FPGA, также известные как мультипроцессоры на программируемых кристаллах – Multiprocessor-on-Programmable Chip, MPoPC или «мультипроцессор с программным ядром» или «мягкий мультипроцессор» – используются не только для прототипирования, но и для окончательного проекта тоже. Рост производительности ПЛИС позволяет разработчикам реализовывать мультипроцессорную систему в

одной интегральной схеме FPGA. Компании, занимающиеся производством FPGA, предлагают возможности применения процессоров, специально разработанных для использования в ПЛИС. Современные FPGA оснащены встроенными блоками памяти, периферийных устройств и межсоединений. Сегодня можно реализовать на ПЛИС мультипроцессорную систему, содержащую 80-100 мягких процессоров [1]. Основным недостатком мультипроцессорных систем на основе FPGA является меньшая производительность по сравнению с ASIC. Однако этот недостаток полностью компенсируется их преимуществами: 1) гибкость и реконфигурация; количество мягких процессоров, которые могут быть включены в проект, ограничивается лишь возможностями FPGA; каждый процессор может быть настроен отдельно путем добавления кэш-памяти, периферийных модулей; 2) сокращение времени выхода на рынок; процесс проектирования не включает этап производства интегральной схемы, что значительно сокращает время разработки проекта; 3) уменьшение затрат; в настоящее время FPGA являются относительно дешевыми, что позволяет выполнять проектирование небольшой группой разработчиков; кроме того, если обнаруживаются ошибки проектирования, их легче и дешевле исправить.

Основными производителями интегральных схем, предназначенных для создания мультипроцессорных систем на основе FPGA, являются компании Xilinx и Altera. Xilinx предлагает два основных процессора: мягкий MicroBlaze и жесткий PowerPC. Количество PowerPC ограничено моделью FPGA и не может быть более четырех. Число MicroBlaze ограничивается только логическими ресурсами и может быть до 80-100 единиц в Virtex-5. Для реализации мультипроцессорных систем компанией Xilinx предложен ряд эффективных широко используемых решений: общая шина для связи ядер CoreConnect On-Chip Peripheral Bus (OPB), и 8 Fast Simple Link (FSL) портов в каждом MicroBlaze для эффективного соединения от точки к точке.

Altera предлагает процессоры Nios и Nios II, шину Avalon и средство проектирования EDA SOPC Builder.

Существуют три основные архитектуры мультипроцессорных систем: Master-Slave, конвейерная (Pipeline) и сетевая (Net). Можно комбинировать архитектуры Master-Slave и конвейерную, что широко используется на практике. В сетевой архитектуре нет иерархии процессоров и все они могут передавать информацию друг другу когда это необходимо. Примером описанной архитектуры является симметричный мультипроцессор (SMP), в котором все процессоры идентичны (однородная мультипроцессорная система). В Master-Slave системе один или несколько процессоров являются управляющими, остальные – управляемыми. Конвейерная архитектура целесообразна для потоковых приложений и представляет собой цепь процессоров, каждый из которых сопоставлен определенному этапу конвейера. Решение задач, разделенное во времени, позволяет повысить производительность системы.

Другим важным вопросом является способ организации связи и его физическая реализация. Существует три подхода: 1) точка-точка, где процессоры подключаются напрямую; преимущество – высокая пропускная способность, потому что нет разделения канала связи, но при расширении системы данный способ не эффективен; 2) общая шина, традиционный подход, пришедший из однопроцессорных систем, который не является эффективным с точки зрения производительности, поскольку в каждый момент времени шина используется только одним процессором; 3) новый и наиболее перспективный подход – сеть-на-кристалле (Network-on-chip, NoC); при наличии большого количества ядер наилучшим образом сочетает площадь кристалла и производительность. Идея заключается в том, чтобы использовать небольшие маршрутизаторы внутри кристалла для обеспечения связи между всеми ядрами системы с низкими задержками.

Известны два способа передачи информации между процессорами: разделяемая память и передача сообщений. Разделяемая память используется наиболее часто, поскольку FPGA имеет ограниченное количество встроенной памяти и данный способ позволяет ее экономить. В системах с разделяемой памятью имеют место проблемы синхронизации и последовательности работы памяти. Как правило, мультипроцессорные системы с общей памятью используют общую шину. Передача сообщений в основном применяется в конвейерных системах.

Существуют 2 вида мультипроцессорных систем: гетерогенные и однородные. Специализированные проблемно-ориентированные системы обычно являются гетерогенными, поскольку они требуют использования различных видов процессоров (мультимедиа, коммуникации, управление, биоинформатика). Однородные MPSoC построены из одинаковых процессоров. В такой системе количество ядер может быть увеличено без изменения архитектуры. Однородные архитектуры, как правило, используются для систем с параллелизмом данных. Например, в беспроводных базовых станциях, где один и тот же алгоритм применяется для нескольких независимых потоков данных.

В настоящее время используются два основных способа создания многопроцессорных систем на базе FPGA: 1) вручную с применением алгоритма, предлагаемого компаниями по производству FPGA; 2) автоматически с помощью технологии мэппинга [31].

2. Восстановление работоспособности систем на кристаллах

Масштабирование наноэлектронных схем и устройств достигло сегодня уровня 28 нм и продолжается дальше. Предполагается, что к 2020 году ведущие компании по производству интегральных схем смогут перейти на технологические нормы 10 нм. Увеличение сложности и производительности устройств на кристаллах неизбежно приводит к появлению проблем, связанных с обеспечением их устойчивости и надежности [7-18]. Выигрыш в площади кристалла и потребляемой мощности, полученный от применения нанотехнологий, компенсируется дополнительными накладными расходами на обнаружение неисправностей и исправление ошибок [19-50].

Обнаружение ошибок путем дублирования и исправление их методом «большинства голосов» – широко распространенные технологии, которые могут существенно увеличивать накладные расходы. В последнее время были предложены более сложные методы, где высокий уровень избыточности применяется выборочно только к критическим сигналам, что позволяет значительно уменьшить стоимость устройства. Дублирование, трехкратное резервирование и обнаружение ошибок с помощью специальных кодов в основном ориентированы на перемежающиеся и постоянные неисправности, возникающие в процессе работы. Использование этих методов не целесообразно, если перемежающиеся неисправности возникают на фоне постоянной ошибки. Такие условия могут возникнуть в интегрированных системах, которые должны надежно работать в течение длительного времени. Если две или три копии работают параллельно, система даже больше подвержена неисправностям из-за индуцированного стрессом влияния износа. Поэтому даже тройное резервирование не является эффективным средством защиты от постоянной ошибки, вызванной износом. Для таких целей используется восстановление работоспособности путем замены дефектных модулей на исправные из числа резервных ресурсов.

Методы встроенного восстановления и самовосстановления разработаны и эффективно применяются для регулярных структур, таких как блоки памяти и программируемые вентильные матрицы, а также для регулярных структур аппаратных средств, цифровых фильтров [7-18]. Однако восстановление нерегулярной логики является трудно реализуемым. Целесообразными представляются дополнительные затраты на восстановление работоспособности переключателей и конфигурационной логики, где минимальный размер основного блока в среднем составляет около 200 транзисторов. Обнаружение неисправности, диагностирование и последующий ремонт путем замены дефектного модуля на исправный резервный требует значительных затрат времени. Указанные операции могут быть выполнены во время переключения системы из штатного режима в режим тестирования и восстановления работоспособности. Для обеспечения ремонта в системе должны находиться в “горячем” состоянии избыточные резервные элементы, средства дублирование/тройного резервирования или обработки ошибок на основе специального кода. Это означает, что для обнаружения ошибок онлайн и выполнения ремонтных операций необходимы дополнительные ресурсы. Добавление в проект нескольких резервных модулей и поддержание их в горячем состоянии требует определенных затрат на протяжении всего жизненного цикла системы, в том числе и дополнительной энергии. Решением данной проблемы может быть архитектура, которая обеспечивает контроль и избыточность только тогда, когда это необходимо. В этом случае проверка ошибок и резервная

избыточность могут быть активизированы по специальным сигналам только при необходимости.

Базовая архитектура, используемая для самовосстановления, показана на рис. 1 [7]. Конфигурируемый блок (RB) состоит из n основных однотипных блоков (BB), которые используются для параллельного выполнения n функций, и одного резервного блока такого же типа. Резервное устройство может быть использовано как для ремонта, так и для переключения функций между функциональными блоками. Набор из двух переключателей и проходных транзисторов в простейшем случае необходим для каждого входа и выхода. Базовая архитектура не содержит никаких дополнительных элементов, таких как компараторы. Однако если переключательная схема модифицирована таким образом, что два блока могут работать параллельно с одинаковыми входными сигналами, то доступно больше функций до тех пор, пока резервный блок не использован для ремонта. В том случае, если основной блок может работать параллельно с двумя другими блоками, например, BasicBlock2 (BB2) с BB1 либо BB3, возможно также диагностирование неисправностей. При автономном тестировании любой из основных блоков может работать параллельно с одним из других блоков. Тогда при работе, например, BB1 и BB2, BB2 и BB3 или BB3 и запасного блока параллельно с одинаковой функцией, неисправный блок может быть обнаружен при автономном тестировании.

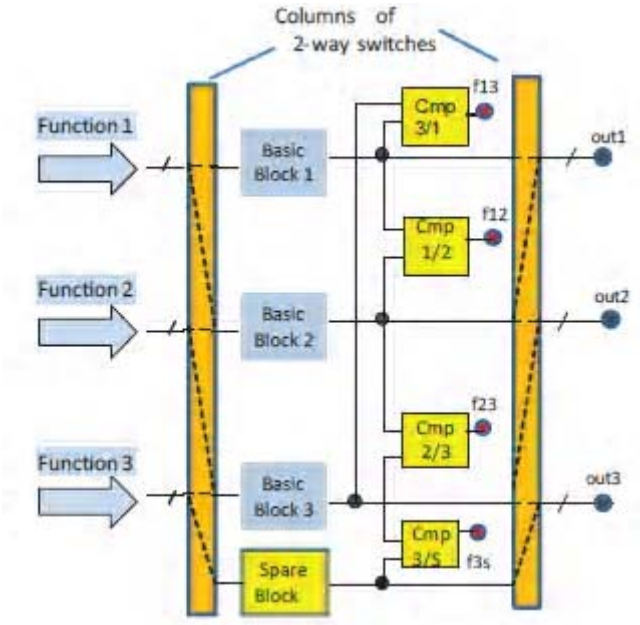


Рис. 1. Реконфигурируемый блок, содержащий один резервный элемент (архитектура 1)

Архитектура со значительно расширенными возможностями показана на рис. 2. Она включает в себя два резервных элемента и трехканальные переключатели на входах и выходах, позволяющие переключать функции прямо, вверх и вниз. Такое переключение и дополнительный резервный блок позволяют запускать две входных функции параллельно на двух основных блоках для проверки онлайн. Кроме того, одна функция может быть выполнена на трех блоках параллельно для проверки онлайн и коррекции ошибок. Два запасных блока могут быть использованы для устранения до двух константных неисправностей, в том числе и производственных дефектов, проявляющихся в процессе эксплуатации.

В обеих описанных выше архитектурах есть два существенных ограничения: отсутствует постоянная онлайн проверка всех сигналов; после обнаружения ошибок процессы диагностирования и ремонта должны быть реализованы в автономном режиме или при запуске, или в течение периода времени, когда система находится «в покое». Усовершенствованная для обеспечения тестирования онлайн и коррекции ошибок архитектура показана на рис. 3.

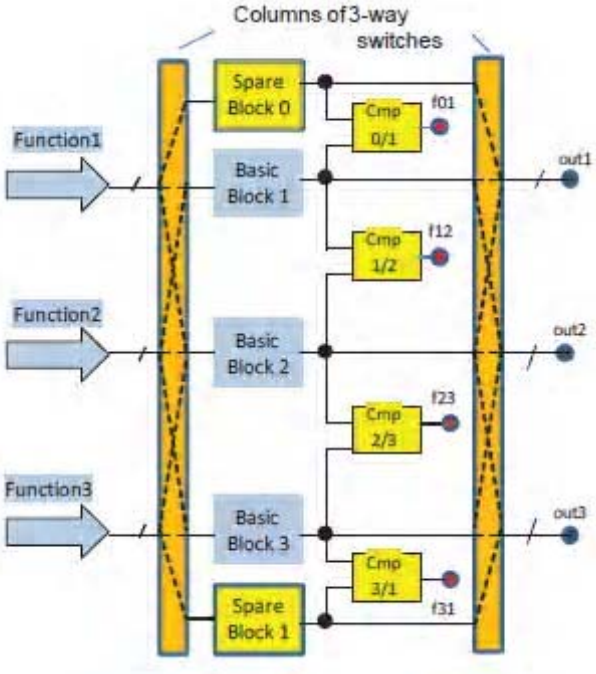


Рис. 2. Реконфигурируемый блок с двумя резервными элементами (архитектура 2)

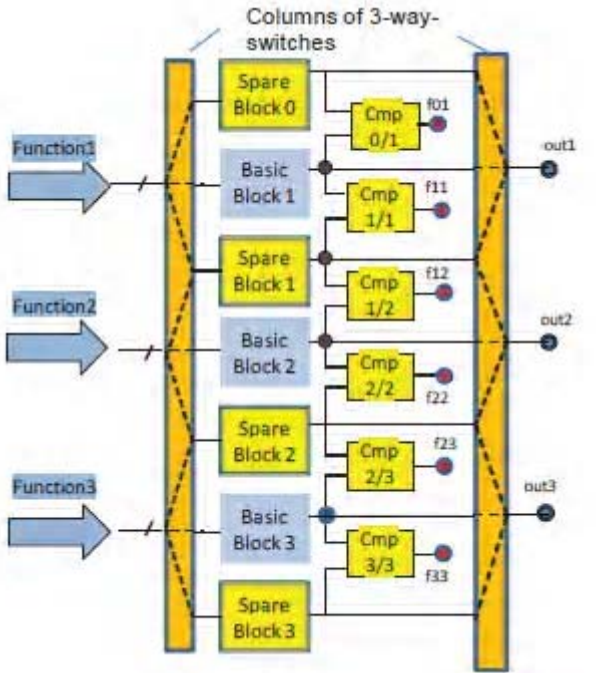


Рис. 3. Реконфигурируемый блок с параллельными резервными элементами (архитектура 3)

Здесь имеется большее количество резервных элементов. Каждый функциональный блок может быть подключен прямо, сверху или снизу. Первый способ функционирования используется только для основных блоков с запасными элементами в режиме ожидания (неактивные) для последующего смещения функций между блоками в целях отключения нагрузки или ремонта. Система с запасными блоками, переведенными в активное состояние для параллельной работы, позволяет выполнять полное обнаружение ошибок онлайн. Вторым режимом работы является обнаружение неисправностей с использованием всех, кроме одного, основных элементов. Один дополнительный запасной блок может использо-

ваться с двойной целью: облегчить параллельную работу после ремонта/замены одного неисправного блока или избирательное тройное резервирование одной функции для выборочной коррекции ошибок.

Наиболее сложная и затратная архитектура показана на рис. 4. Здесь все функции могут работать при тройном резервировании для обнаружения ошибок онлайн. Четыре переключателя для входа/выхода обеспечивают избирательное тройное резервирование (Triple modular redundancy, TMR) даже при наличии одиночной или кратной константной неисправности.

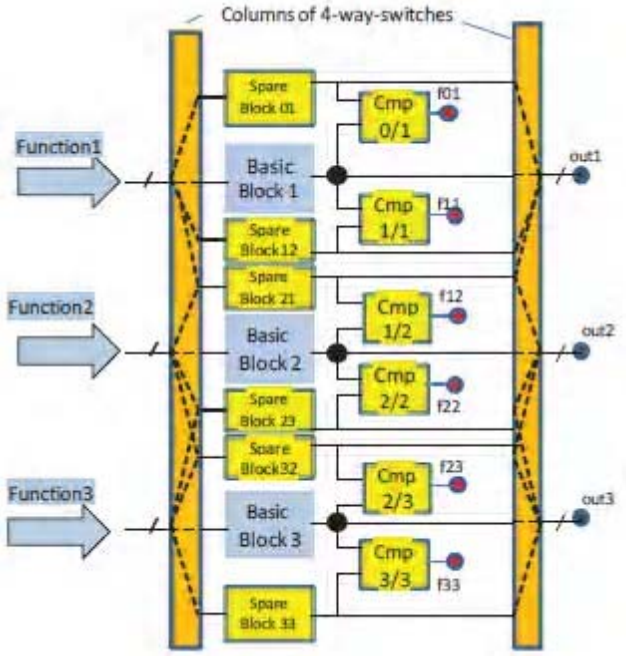


Рис. 4. Реконфигурируемый блок с двойными параллельными резервными элементами (архитектура 4)

Тройное резервирование является наиболее распространенной архитектурой для обнаружения ошибок онлайн и их исправления. Однако для многих приложений связанные с этим накладные расходы непомерно высоки. Избирательное TMR для конкретных сигналов является одним из вариантов решения данной проблемы. Для многих типов интегрированных систем можно ввести классификацию сигналов в зависимости от временных ограничений с одной стороны и уровня безопасности критической функциональности – с другой. В работе выделены три уровня сигналов (рис. 5): 1) нет жестких временных ограничений, безопасность не критична; 2) нет жестких временных ограничений, безопасность критична; 3) есть жесткие временные ограничения, безопасность критична.

Быстрая реакция означает, что необходим отклик в пределах одного такта. Только сигналы уровня 3 требуют постоянного контроля в виде тройного резервирования; для сигналов уровней 1и 2 TMR не требуется. Например, сигналы уровня 2 могут допускать небольшую задержку с последующей коррекцией ошибки. Такое поведение необходимо для перемежающихся и константных неисправностей, появляющихся во время работы устройства. Только константные неисправности требуют дополнительного процесса самовосстановления с использованием избыточных ресурсов, который должен быть выполнен в автономном режиме.

Архитектуры типов 2 или 3 допускают селективное TMR при возникновении неисправности. В архитектуре 3 обнаружение неисправностей осуществляется путем дублирования и сравнения. Добавление третьего элемента выполняется по требованию после обнаружения неисправности с последующей TMR-операцией. По сути, дополнительное время тратится в обмен на дополнительное оборудование (архитектура 3 против 4). При этом принципиальная схема требует некоторых изменений (дополнительные накладные расходы). В случае появления ошибки она должна быть исправлена и функциональный выход сначала должен

быть изолирован. Для выполнения данной операции может быть использован С-элемент Мюллера (Muller-C-element) (рис. 6).

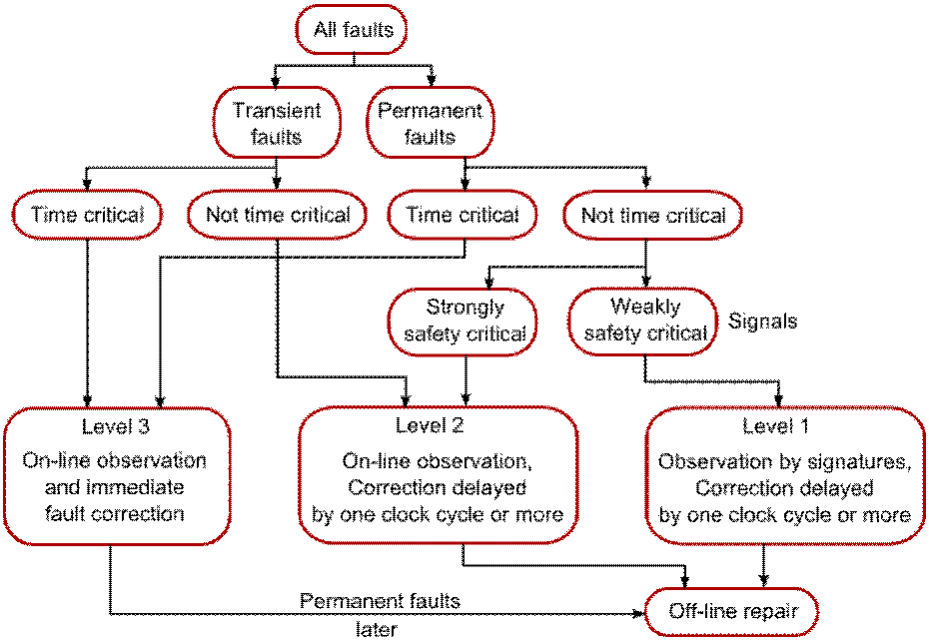


Рис. 5. Классификация сигналов и операций восстановления

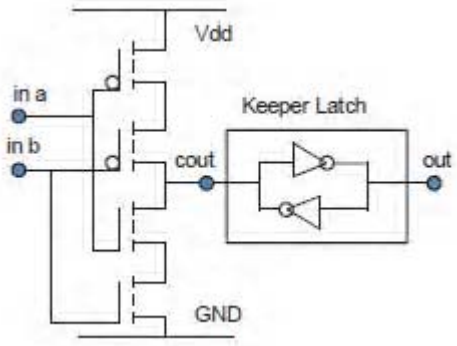


Рис. 6. С-элемент Мюллера

Если входы a и a отличаются, выход cout отделяется от Vdd и GND и становится плавающим. Тогда запорная защелка (keeper latch) может быть добавлена для создания определенного логического уровня на выходе. Избирательная схема, основанная на С-элементе Мюллера, показана на рис. 7. Например, имеется неисправность в базовом блоке 1. Три основных блока работают параллельно, и только С-элемент Мюллера, объединяющий выходы SpareBlock0 и SpareBlock1, определяет значение на выходе, в то время как другие находятся в состоянии высокого импеданса. Значение на выходе out устанавливается правильным выходом. Основные архитектуры 1, 2 и 3 могут быть модифицированы путем добавления С-элементов Мюллера на выходы схем.

Режим работы, основанный на архитектуре 3, описан ниже. При нормальной работе два функциональных блока работают параллельно. Их выходы объединены С-элементом Мюллера и подвергаются сравнению. Если значения на выходах различаются, С-элемент Мюллера отделяет результаты от выходов, и компаратор сигнализирует о неисправности. На выходе схемы может быть восстановлено предыдущее значение с помощью запорной защелки (см. рис. 6). При появлении сигнала неисправности соответствующая функция переключается на TMR за счет соседних функций, которые затем работают как единый функциональный модуль.

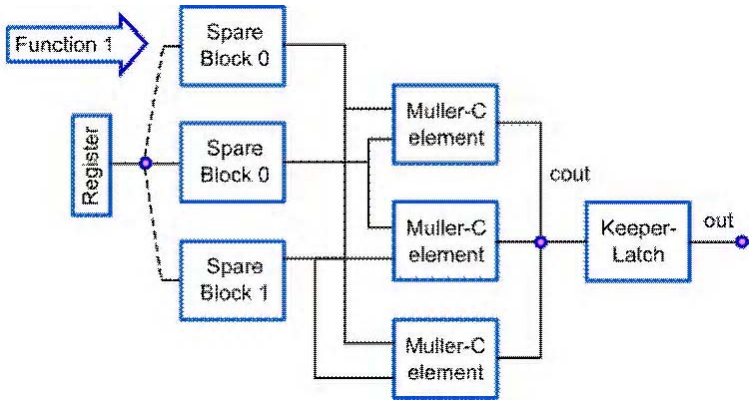


Рис. 7. Выбор с помощью С-элемента Мюллера

Набор из трех С-элементов Мюллера, предназначенный для сравнения выходов двух функциональных блоков, подключается к выходу cout. Только значения на выходах правильно функционирующих функциональных блоков транспортируются на выход схемы (рис. 8).

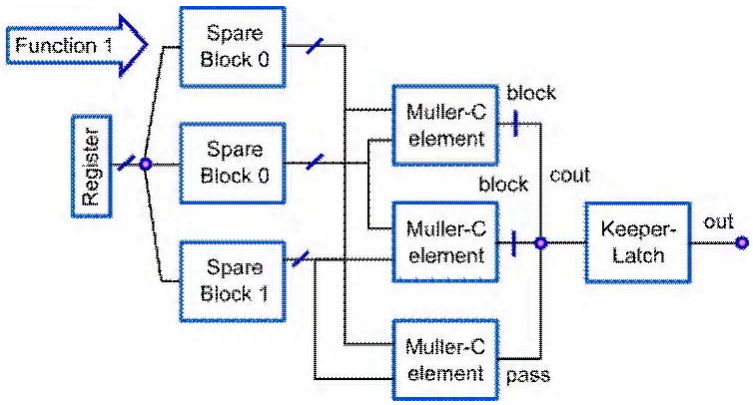


Рис. 8. Избирательное TMR при наличии неисправности

Все операции могут выполняться без прерывания, «правильный» выход дефектной функции появится с некоторой задержкой, которая может сохраняться в течение такта. Расширенная архитектура реконфигурируемого блока, представленного на рис. 2, показана на рис. 9.

В результате исследований [7-9] было определено, что чем больше выбирается размер блоков, на которые разбивается система, тем меньше процент избыточной аппаратуры, необходимой для реализации схемы самотестирования и восстановления. Для 32-битового АЛУ число избыточных элементов равняется 38%, в то время как для вентиля И-НЕ – 200%.

В работах [7-9] упоминается FPGA, но нет привязки к реальной архитектуре программируемой логической матрицы и конфигурируемым блокам CLB, которые являются основными элементами для реализации комбинационной части устройств. Замена одного логического элемента в рамках CLB невозможна, поскольку функция на блоках реализуется в табличном виде. Минимальным блоком для построения самотестируемых схем может быть только CLB.

Описанный метод самотестирования неприменим к отдельным блокам сумматора и устройствам, реализованным на суммировании. Сумматоры строятся на сквозных линиях переноса, идущих вдоль столбцов CLB в FPGA. Исключить из сумматора блок, не нарушив целостности всего устройства, невозможно. Можно выполнять резервирование только для целого компонента. Создание схемы самотестирования на уровне логических элементов приводит к существенной избыточности аппаратуры. Для более высокого уровня сложно

получить однородную схему, которую можно было бы преобразовать в самовосстанавливающуюся систему. В работах [7-9] упоминается метод самовосстановления, основанный на регулярной структуре FPGA, который предлагает исключать из использования столбец или строку CLB в случае обнаружения в нем неисправного блока. Несмотря на то, что метод, как считают авторы, требует улучшения, он является более реальным и технологичным [14]. Кроме того, может быть выполнено резервирование компонентов на уровне столбцов CLB путем внесения дополнительной избыточной теневой конфигурации.

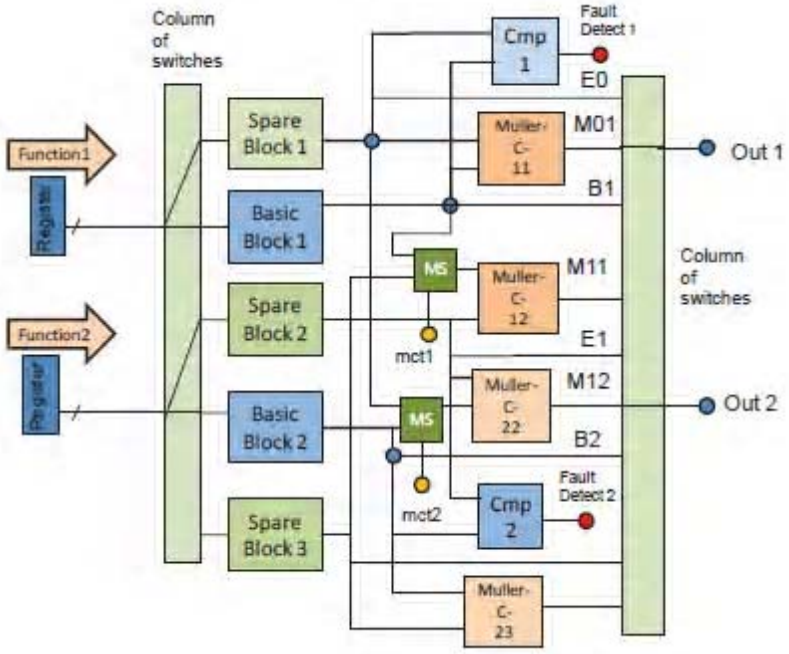


Рис. 9. Реконфигурируемый логический блок с компараторами и С-элементами Мюллера

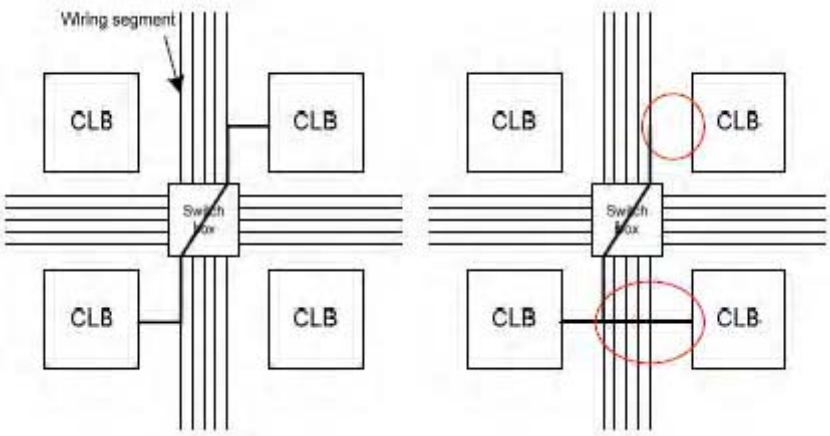
Встроенный метод восстановления работоспособности мультипроцессорных систем на кристалле предложен в [18]. Ионизирующее излучение может вызывать нежелательные эффекты в полупроводниковых устройствах, например, переключение состояния ячеек памяти (flipping), которые называют мягкими ошибками и относят к распространенному типу нарушений одного события (single event upsets, SEU).

Программируемые логические матрицы FPGA на основе SRAM восприимчивы к неисправностям SEU, при которых нежелательное событие может приводить к однократному кратковременному переключению бита элемента памяти FPGA.

Неисправности SEU в FPGA могут появляться в ячейках встроенной и конфигурационной памяти. Неисправности встроенной памяти повреждают содержимое блока RAM или триггера. Ошибки конфигурационной памяти являются наиболее сложными, поскольку они изменяют поведение FPGA, определяемое набором конфигурируемых логических блоков (Configurable Logic Blocks, CLB) и их межсоединениями.

Межсоединения организованы с помощью переключательных блоков и сегментов трасс, как показано на рис. 10, а. Состояние переключательного блока и сегментов трасс определяется логическим состоянием их конфигурационных битов. Ошибки SEU, влияющие на указанные биты, могут отключить или неправильно подключить сегмент проводника, который соединяет блоки CLB (рис. 10, б).

Упрощенная структура CLB Xilinx FPGA показана на рис. 11. Конфигурируемый блок состоит из набора таблиц преобразования (Look-up Tables – LUTs), триггеров и межсоединений. Источниками неисправностей SEU в CLB являются: 1) биты таблицы преобразований, когда SEU изменяет логическую функцию, имплементированную в look-up table; 2) конфигурационные биты таблицы преобразований; например, биты, устанавливаемые, если ресурс look-up table конфигурирован как таблица преобразований, двойной порт RAM или регистр сдвига; SEU изменяет функциональность таблицы преобразований; 3) мультиплексоры и инверторы, где SEU изменяет внутренние связи CLB.



а б

Рис. 10. Межсоединения FPGA

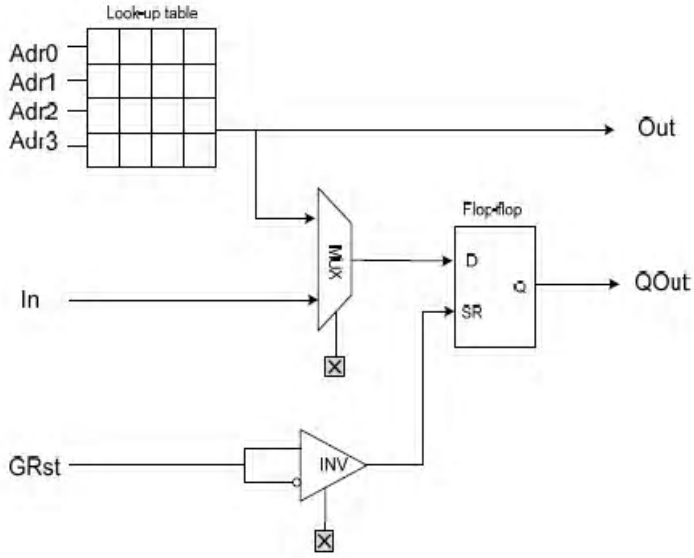


Рис. 11. Источники неисправностей в структуре CLB

Метод коррекции ошибок мультипроцессорных систем на базе FPGA описан ниже. Одно из процессорных ядер сканирует конфигурационные фреймы и выполняет реконфигурацию в случае обнаружения неисправностей (первый проход). Если неисправность появилась непосредственно в процессорном ядре, другое ядро принимает на себя функцию восстановления и выполняет реконфигурацию (расширенное восстановление). Во время нормальной работы системы выбранные процессорные ядра выполняют проверку конфигурации параллельно с выполнением целевого системного приложения. Схемы FPGA являются уязвимыми для SEU и неисправность может возникнуть в любой ячейке конфигурационной памяти в любое время.

Обнаружение и исправление ошибок производится с помощью периферийных ядер с внутренним доступом к конфигурации памяти FPGA. В ПЛИС фирмы Xilinx используется ядро Internal Configuration Access Port (ICAP), которое позволяет встроенному микропроцессору выполнять операции чтения и записи в конфигурационной памяти FPGA с помощью ICAP в реальном времени. Это дает возможность пользователю создавать программное обеспечение для встроенных процессоров в целях изменения структуры и функциональности схемы в процессе ее функционирования. Устройства FPGA Virtex 4 и Virtex 5 включают

в себя средства коррекции ошибок Error Correcting Code (ECC), которые могут быть использованы также для обнаружения ошибок в конфигурационных фреймах. ECC применяют алгоритм кода Хемминга, который определяет местонахождение одиночной неисправности и обнаруживает кратные ошибки. После считывания информации из конфигурационного фрейма ECC формирует значение синдрома (syndrome value). Для одиночной ошибки 11-битное значение синдрома идентифицирует ошибочный бит фрейма.

Возможные сценарии появления ошибок и их исправления:

– ошибка возникает на процессоре, реализующем функциональность, или периферийном модуле; проверяющий процессор корректирует ошибку;

– кратная ошибка возникает на процессоре, реализующем функциональность, или периферийном модуле; неисправность может быть обнаружена, но не может быть исправлена; выполняется реконфигурирование всей системы;

– ошибка останавливает работу проверяющего процессора; после блокировки по времени сторожевой таймер вызывает следующий процессор, который пытается исправить ошибку;

– проверяющий процессор сигнализирует о неудачном выполнении процедуры самотестирования; процессор останавливается; после блокировки по времени сторожевой таймер вызывает следующий процессор, который пытается исправить ошибку;

– ошибка влияет на синхронизацию по времени; сторожевой таймер обнаруживает ошибку, которая не может быть исправлена; выполняется реконфигурирование всей системы;

– ошибка влияет на глобальные сигналы и конфигурационные регистры FPGA; сторожевой таймер обнаруживает ошибку, которая не может быть исправлена; выполняется реконфигурирование всей системы;

– аппаратная ошибка, постоянная неисправность схемы, которая не может быть устранена путем реконфигурации; в случае обнаружения одиночной ошибки запускается алгоритм восстановления работоспособности фрейма, если восстановление не выполнено, то выдается сообщение об аппаратной ошибке.

Аппаратная архитектура приведена на рис. 12. Она включает в себя следующие компоненты: мультиядерная система с n встроенными микропроцессорными ядрами; внутренний интерфейс частичной реконфигурации, общий доступ к которому имеют все процессорные ядра; механизм взаимного исключения для обеспечения возможности исключающего доступа к разделяемым периферийным модулям; контроллер прерываний; внешний сторожевой таймер.

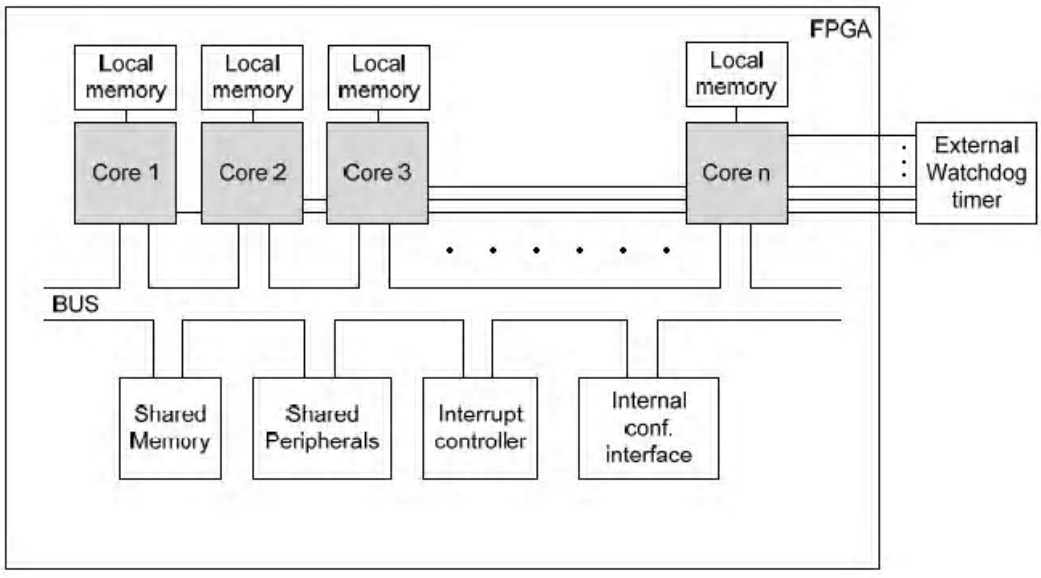


Рис. 12. Архитектура аппаратной платформы

3. Заключение

Выполненный обзор технологий восстановления работоспособности мультипроцесорных систем на кристаллах позволяет определить спектр задач дальнейшего совершенствования инфраструктуры встроенного сервисного обслуживания цифровых систем на кристаллах в целях уменьшения времени восстановления работоспособности, что позволяет существенно повысить качество изделия или выход годной продукции. Необходимо разработать метод синтеза функциональных описаний, ориентированных на имплементацию в кристаллы PLD, модели оценивания качества транзакционных соединений в архитектуре вычислительного устройства, автоматную модель переадресации дефектных компонентов комбинационной схемы путем использования ремонтного запаса примитивных элементов, специализированный процессор обработки нечисловых данных, который характеризуется ограниченным набором векторных логических операций, что дает возможность на порядок повысить быстродействие процедур тестирования функциональных нарушений цифровых изделий.

Дальнейшие исследования направлены на повышение качества цифровых изделий за счет встроенного сервисного обслуживания, включающего автономное восстановление работоспособности на основе кубитных моделей функциональностей и специальных технологий диагностирования и ремонта, которые позволяют получить минимальное количество дефектных компонентов цифровой системы на кристалле и минимальное время восстановления работоспособности благодаря нефункциональной программно-аппаратной избыточности, формирующей инфраструктуру SoC.

Список литературы: 1. *Dorta T.* Overview of FPGA-Based Multiprocessor Systems / T. Dorta, J. Jimenez, J.L. Martin, U. Bidarte, A. Astarloa // 2009 Intern. Conf. on Reconfigurable Computing and FPGAs. 2009. P. 273 – 278. **2.** *Sterpone L.* A New Fault Injection Approach for Testing Network-on-Chips / L. Sterpone, D. Sabena, M.S. Reorda // 20th Euromicro International Conf. on Parallel, Distributed and Network-Based Processing. 2012. P. 530 – 535. **3.** *Sharma M.* A novel Test Access Mechanism for failure diagnosis of multiple isolated identical cores / M. Sharma, A. Dutta, Wu-Tung Cheng, B. Benware, M. Kassab // IEEE International Test Conference (ITC). 2011. P. 1 – 9. **4.** *Chakrabarty K.* Toward Bug-free Multicore SoC Architectures: System Validation with Transaction-Level Models / K. Chakrabarty // IEEE Design & Test of Computers.– 2011. Vol. 28, Issue 3. P. 4. **5.** *Mishra P.* Guest Editors' Introduction:Multicore SoC Validation withTransaction-Level Models / Prabhat Mishra, Sandeep Shukla, Zeljko Zilic // IEEE Design & Test of Computers. 2011. Vol. 28, Issue 3. P. 6-7. **6.** *Wagner I.* Reversi: Post-silicon validation system for modern microprocessors / I. Wagner, V. Bertacco // Computer Design. ICCD. Oct. 2008.P.307-314. **7.** *Koal, T.; Ulbricht, M.; Vierhaus, H.T.* Combining on-line fault detection and logic self repair / Koal, T.; Ulbricht, M.; Vierhaus, H.T. // IEEE 15th Intern. Symposium on Design and Diagnostics of Electronic Circuits & Systems (DDECS). 2012. P. 288 – 293. **8.** *Koal T. A* Concept for Logic Self Repair / T. Koal, D. Scheit, Vierhaus H.T. // 12th Euromicro Conference on Digital System Design. Architectures, Methods and Tools. 2009. P. 621 – 624. **9.** *Koal T.* Basic Architecture for Logic Self Repair / T. Koal, H.T. Vierhaus // 14th IEEE Intern. On-Line Testing Symposium. 2008. P. 177 – 178. **10.** *Habermann S.* Built-in Self Repair by Reconfiguration of FPGAs / S. Habermann, R. Kothe, H.T. Vierhaus // 12th IEEE International On-Line Testing Symposium. 2006. **11.** *Koal T.* Optimal spare utilization for reliability and mean lifetime improvement of logic built-in self-repair / T. Koal, H.T. Vierhaus // 14th International Symposium Design and Diagnostics of Electronic Circuits & Systems (DDECS). 2011. P. 219 – 224. **12.** *Goldstein S.* Reconfigurable computing and electronic nanotechnology / S. Goldstein, M. Budiu, M. Mishra, G. Venkataramani // Proc. of the IEEE International Conference on Application-Specific Systems, Architectures, and Processors. 2003. P. 132 – 142. **13.** *Lei Sun.* Runtime Self-Diagnosis and Self-Recovery Infrastructure for Embedded Systems / Lei Sun, Yuki Kinebuchi, Tomohiro Katori, Tatsuo Nakajima // Third IEEE International Conference on Self-Adaptive and Self-Organizing Systems. 2009. P. 284 – 285. **14.** *Rab M.T.* Improving Memory Repair by Selective Row Partitioning / M.T. Rab, A.A. Bawa, N.A. Touba // 24th IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems. 2009. P. 211 – 219. **15.** *Vierhaus H.T.* Embedded Diagnostic Logic Test Exploiting Regularity / H.T. Vierhaus, R. Kothe // 11th EUROMICRO Conference on Digital System Design Architectures, Methods and Tools. 2008. P. 873 – 879. **16.** Koal T. A scheme of logic self repair including local interconnects / T. Koal, D. Scheit, H.T. Vierhaus // 12th International Symposium on Design and Diagnostics of Electronic Circuits & Systems (DDECS '09). 2009.P. 8 – 11. **17.** *Koal T.* On the Feasibility of Built-In Self Repair for Logic Circuits / T. Koal, D. Scheit, M. Schцlzel, H.T. Vierhaus // IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT). 2011. P. 316 – 324. **18.** *Legat U.* On line self recovery of embedded multiprocessor SOC on FPGA using dynamic partial reconfiguration / Uros Legat, Anton Biasizzo, Franc Novak / / Information Technology And Control. 2012. Vol. 41. No 2. P. 116-124. **19.** *Zorian Yervant.* Tutorial on

EWDTS 2007. Embedding Infrastructure IP for SOC Yield Improvement / Yervant Zorian // Proceedings of the 39th Design Automation Conference.– New Orleans, LA, USA. 2002. P. 709-712. **20.** *Zorian Yervant.* Test Strategies for System-in-Package / Yervant Zorian // The Plenary Paper of IEEE East-West Design & Test Symposium (EWDTS'08). Lviv, Ukraine. October 9-12, 2008. **21.** Bhattacharya N. SoftwareHardware Hybrid Systems Verification / N. Bhattacharya // Software Testing, Verification and Validation (ICST). 2011. P.435-438. **22.** *Zilic Z.* Challenges of Rapidly Emerging Consumer Space Multiprocessors / Z. Zilic, P. Mishra, S.Shukla // Design & Test of Computers. May-June 2011. Vol. 28, No. 3. P.52-53. **23.** *Kularatna N.* Electronic circuit design from concept to implementation / N. Kularatna. CRC press NW. 2008. 504 p. **24.** *Abramovici M.* BIST-Based Delay-Fault Testing in FPGAs / Miron Abramovici, Charles E. Stroud // Journal of Electronic Testing: Theory & Applications. 2003. Vol. 19, No. 5. P. 549-558. **25.** *Ulbricht M.* A new hierarchical built-in self-test with on-chip diagnosis for VLIW processors / Markus Ulbricht, Mario Scholzel, Tobias Koal, Heinrich Theodor Vierhaus // Design and Diagnostics of Electronic Circuits & Systems (DDECS). April 2011. P.143-146. **26.** *Elm M.* BISD: Scan-based Built-In self-diagnosis / M. Elm, H.-J. Wunderlich // Design, Automation & Test in Europe Conference & Exhibition (DATE). March 2010. P. 1243-1248. **27.** *Huang Yu-Jen.* A built-in self-test scheme for the post-bond test of TSVs in 3D ICs / Yu-Jen Huang, Jin-Fu Li, Ji-Jan Chen, Ding-Ming Kwai, Yung-Fa Chou, Cheng-Wen Wu // VLSI Test Symposium (VTS). May 2011. P.20-25. **28.** Shianling Wu. Logic BIST Architecture Using Staggered Launch-on-Shift for Testing Designs Containing Asynchronous Clock Domains / Wu Shianling, Laung-Terng Wang, Yu Lizhen, H. Furukawa, Wen Xiaoqing, W.-B. Jone, N.A. Touba, Zhao Feifei, Liu Jinsong, Hao-Jan Chao, Li Fangfang, Jiang Zhigang // Defect and Fault Tolerance in VLSI Systems (DFT). Oct. 2010. P. 358-366. **29.** *Tsu-Wei Tseng.* A Shared Parallel Built-In Self-Repair Scheme for Random Access Memories in SOCs / Tsu-Wei Tseng, Jin-Fu Li // Test Conference. ITC 2008. Oct. 2008. P.1-9. **30.** *Novak O.* Handbook of testing electronic systems / O. Novak, E. Gramatova, R.Ubar. Czech University Publishing House. 2005. 402 p. **31.** *Проектирование* и тестирование цифровых систем на кристаллах / В.И. Хаханов, Е.И. Литвинова, О.А. Гузь. Харьков: ХНУРЭ. 2009. 484с. **32.** *Richter K.* A formal approach to MpSoC performance verification / K. Richter, M. Jersak, R. Ernst // Computer. 2003. Vol. 36, Iss. 4. P. 60 – 67. **33.** *Cordibella S.* A HW/SW co-simulation framework for the verification of multi-CPU systems / S. Cordibella, F. Fummi, G. Perbellini, D. Quaglia // IEEE Intern. High Level Design Validation and Test Workshop. 2008. P. 125 – 131. **34.** *Khan G.N.* Simulation environment for design and verification of Network-on-Chip and multi-core systems / G.N. Khan, V. Dumitriu // IEEE International Symposium on Modeling, Analysis & Simulation of Computer and Telecommunication Systems. 2009.P. 1 – 9. **35.** *Хаханов В.И., Хаханова И.В., Литвинова Е.И., Гузь О.А.* Проектирование и верификация цифровых систем на кристаллах. Verilog & System Verilog: Харьков: Новое слово, 2010. 528с. **36.** Основы технической диагностики / Под. ред. П.П.Пархоменко. М.: Энергия, 1976. 460с. **37.** *Пархоменко П.П.* Основы технической диагностики (Оптимизация алгоритмов диагностирования, аппаратурные средства) / П.П. Пархоменко, Е.С. Согомонян / Под ред. П.П. Пархоменко.М.: Энергия, 1981. 320 с. **38.** *Da Silva F.* The Core Test Wrapper Handbook. Rationale and Application of IEEE Std. 1500™ / F. Da Silva, T. McLaurin, T. Waayers. Springer.2006.XXIX.276 p. **39.** *Marinissen E.J.* Guest Editors' Introduction: The Status of IEEE Std 1500 / E.J. Marinissen, Yervant Zorian // IEEE Design & Test of Computers. 2009. No26(1). P.6-7. **40.** *Ubar R.* Embedded diagnosis in digital systems / R. Ubar, S. Kostin, J. Raik // 26th International Conference "Microelectronics", MIEL 2008. 2008. P. 421-424. **41.** *Elm M.* Scan Chain Organization for Embedded Diagnosis / M. Elm, H.-J. Wunderlich // Design, Automation and Test in Europe, DATE '08. 2008. P. 468–473. **42.** *Автоматизация* диагностирования электронных устройств/ Ю.В.Малышенко и др./ Под ред. В.П.Чипулиса. М.: Энергоатомиздат, 1986. 216с. **43.** *Хаханов В.И.* Встроенное диагностирование цифровых систем / В.И. Хаханов, С.В. Чумаченко, Y. Tiecoura, С.С. Галаган // Радіоелектронні і комп'ютерні системи. 2009. №7(41). С. 314-318. **44.** *Hahanov V.I.* Embedded Testing for SoC Functionality // V.I. Hahanov, S. Pokrova, Y. Tiecoura, A.A. Gorobets // Proc. of the Xth International Conference CADSM. Lviv-Polyana. 2009. P. 29-33. **45.** *Bing-Yang Lin.* A Memory Failure Pattern Analyzer for memory diagnosis and repair / Lin Bing-Yang, Mincent Lee, Cheng-Wen Wu // IEEE 30th VLSI Test Symposium (VTS). 2012. P. 234 – 239. **46.** *Sha'afi Kabiri P.* Effective RT-level software-based self-testing of embedded processor cores / P. Sha'afi Kabiri, Z. Navabi, // IEEE 15th International Symposium on Design and Diagnostics of Electronic Circuits & Systems (DDECS). 2012. P. 209 – 212. **47.** *Tsertov A.* Automatic SoC Level Test Path Synthesis Based on Partial Functional Models / A. Tsertov, R. Ubar, A. Jutman, S. Devadze // 20th Asian Test Symposium (ATS). 2011. P. 532 – 538. **48.** *Grosso M.* A software-based self-test methodology for system peripherals / M. Grosso, W.J. Perez H, D. Ravotto, E. Sanchez, M.S. Reorda, J.V. Medina // 15th IEEE European Test Symposium (ETS). 2010. P. 195 – 200. **49.** *Haghbayan M.H.* Architecture design and technical methodology for bus testing / M.H. Haghbayan, Z. Navabi // East-West Design & Test Symposium (EWDTS).2010. P. 504 – 509. **50.** Ecker W. TLM+ modeling of embedded HW/SW systems / W. Ecker, V. Esen, R. Schwencker, T. Steininger, M. Velten // Conference & Exhibition Design, Automation & Test in Europe (DATE). 2010. P. 75 – 80.

Поступила в редколлегию 19.06.2012

Мурад Али Аббас, аспирант кафедры АПВТ ХНУРЭ. Научные интересы: техническая диагностика цифровых систем и сетей. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-326.

Багхдади Аммар Авни Аббас (Baghdadi Ammar Awni), аспирант кафедры АПВТ ХНУРЭ, Baghdad University. Научные интересы: техническая диагностика цифровых систем и сетей. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-326.

Хаханов Владимир Иванович, декан факультета КИУ ХНУРЭ, д-р техн. наук, 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.

Литвинова Евгения Ивановна, д-р. техн. наук, профессор кафедры АПВТ ХНУРЭ. Научные интересы: автоматизация диагностирования и встроенный ремонт компонентов цифровых систем в пакете кристаллов. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-421. E-mail: kiu@kture.kharkov.ua.

Дахири Фарид, студент факультета КИУ ХНУРЭ. Научные интересы: техническая диагностика цифровых систем и сетей, программирование мобильных платформ. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-326.

УДК 621.372.061

А.М. ЗЕМЛЯК, Т.М. МАРКИНА

ХАРАКТЕРИСТИКИ РАЗЛИЧНЫХ СТРАТЕГИЙ ПРОЕКТИРОВАНИЯ АНАЛОГОВЫХ ЦЕПЕЙ В РАСШИРЕННОМ БАЗИСЕ

Методология проектирования аналоговых цепей, разработанная ранее на основе применения теории оптимального управления, обобщается в целях формирования более полного структурного базиса различных стратегий проектирования. Это приводит к значительному увеличению возможных стратегий проектирования и к появлению более перспективных стратегий, позволяющих еще больше сократить время оптимизации цепи. Численные результаты показывают возможность существенного сокращения процессорного времени, необходимого для оптимизации цепей.

1. Введение

Решение задачи сокращения времени проектирования электронных цепей позволит улучшить характеристики проектирования. В работах [1,2] авторы утверждают, что отказ от соблюдения законов Кирхгофа в процессе оптимизации цепи позволил существенно сократить время проектирования. Впервые эта идея, применительно к электронным цепям, была высказана в работе [3]. Задача проектирования на основе формулировки процесса проектирования в терминах теории оптимального управления обобщена в [4–6]. При этом задача оптимального по времени проектирования формулируется как типичная задача минимизации функционала в теории управления. Функционалом, подвергаемым минимизации, является процессорное время. Теория, развитая в этом подходе, базируется на том, что переменные цепи, определенные первоначально как зависимые, т.е. узловые напряжения, могут объявляться независимыми. При этом, однако, переменные, первоначально определенные как независимые, таковыми остаются в процессе проектирования.

2. Формулировка задачи

Переопределение бывших зависимых переменных как независимых, а также возможное их обратное переопределение производится путем введения специального управляющего вектора U, имеющего размерность, равную числу зависимых переменных, которые совпадает с числом узлов схемы. Модель цепи в этом случае описывается следующей системой уравнений [4]:

$$(1-u_j)g_j(X)=0,\ j=1,2,...,M, \qquad (1)$$

где $U = (u_1, u_2, ..., u_M)$, $u_j \in \Omega$, $\Omega = \{0,1\}$, M – число зависимых переменных, совпадающее с числом узлов схемы. В то же время процедура оптимизации цепи может быть задана следующей системой уравнений [5]:

$$x_i^{s+1} = x_i^s + t_s \cdot f_i(X,U),\ i = 1,2,...,N. \qquad (2)$$

Здесь N – число всех переменных, а правые части $f_i(X,U)$ задаются конкретным методом оптимизации и, например, для градиентного метода определяются следующими выражениями [5]:

$$f_i(X,U) = -b\frac{\delta}{\delta x_i}\{C(X) + \frac{1}{\varepsilon}\sum_{j=1}^{M} u_j g_j^2(X)\},\ i = 1,2,...,K, \qquad (3)$$

$$f_i(X,U) = -b \cdot u_{i-K}\frac{\delta}{\delta x_i}\{C(X) + \frac{1}{\varepsilon}\sum_{j=1}^{M} u_j g_j^2(X)\} + \frac{(1-u_{i-K})}{dt}\{-x'_i + \eta_i(X)\},$$

$$i = K+1, K+2, ..., N,$$

где K – число независимых переменных в традиционной постановке задачи; $C(X)$ – целевая функция процесса проектирования, функция $\eta_i(X)$, записанная в неявном виде, определяет текущее значение переменной x_i, $x_i = \eta_i(X)$; x'_i – предыдущее значение переменной x_i.

Как показано в [4,5], в этом случае появляется множество различных стратегий проектирования, представляющих собой структурный базис, который включает 2^M различных стратегий. Этот базис является основой для поиска стратегий, являющихся более быстрыми по сравнению с традиционной стратегией проектирования. В работе [6] были найдены стратегии, позволяющие во много раз сократить время оптимизации электронной цепи по сравнению с традиционной стратегией.

Однако построенный базис не является полным, так как переменные цепи не уравнены в правах, поскольку только исходные зависимые переменные могут стать независимыми, но исходные независимые переменные не могут стать зависимыми. В то же время можно уравнять в правах все переменные, т. е. можно считать, что любая переменная может быть объявлена независимой или зависимой на любом шаге процедуры оптимизации электронной цепи. В этом случае требуется изменить уравнение модели цепи. Уравнение (1), определяющее модель цепи, преобразуется в следующее:

$$(1-u_i)g_j(X) = 0,\ i = 1,2,...,N,\ j \in J, \qquad (4)$$

где J есть множество индексов для тех функций $g_j(X)$, для которых $u_i = 0$, $J = \{j_1, j_2, ..., j_Z\}$, $j_s \in \Pi$ при s = 1, 2, . . ., Z, Π есть множество индексов от 1 до M, $\Pi = \{1,2,...,M\}$, Z есть число уравнений, которые остаются в системе (4), т.е. $Z \in \{0,1,...,M\}$, и управляющий вектор состоит из N компонент $U = (u_1, u_2, ..., u_N)$.

Правые части системы (2) определяются при этом следующими выражениями:

$$f_i(X,U) = -b \cdot u_i\frac{\delta}{\delta x_i}F(X,U) + \frac{(1-u_i)}{dt}\{-x_i(t-dt) + \eta(X)\},\ i = 1,2,...,N, \qquad (5)$$

здесь $F(X,U)$ есть обобщенная целевая функция, определяемая формулой:

$$F(X,U) = C(X) + \frac{1}{\varepsilon}\sum_{j \in \Pi \setminus J} u_j g_j^2(X). \qquad (6)$$

Подобное определение процесса проектирования является более общим по сравнению с данным в [4–7]. Ранее определенную методологию назовем соответствующей первому уровню обобщения и новую, представленную в данной работе, соответствующей второму уровню обобщения. Новый уровень обобщения производит более представительный структурный базис различных стратегий проектирования. Полное число различных стратегий

проектирования, составляющих новый структурный базис, определяется суммой различных комбинаций и равно $\sum_{i=0}^{M} C_{K+M}^{i}$. Использование этого расширенного структурного базиса открывает новые возможности при поиске перспективных стратегий.

3. Численные результаты

Проанализируем примеры проектирования некоторых электронных цепей на основе нового структурного базиса. Целевая функция $C(X)$ определена как сумма квадратов разностей между заранее определенными и текущими значениями узловых напряжений для некоторых узлов.

Рассмотрим задачу проектирования пассивной электронной цепи с двумя узлами, изображенной на рис. 1 и представляющей собой простой нелинейный делитель напряжения.

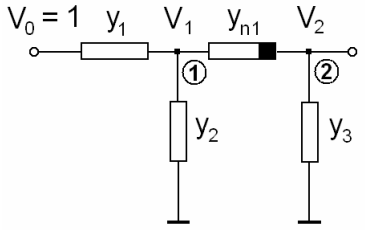


Рис. 1. Схема с тремя независимыми (K=3) и двумя зависимыми (M=2) переменными

Нелинейный элемент определяется следующей зависимостью: $y_{n1} = y_0 + b(V_1 - V_2)^2$. Вектор X включает пять компонент: $x_1^2 = y_1$, $x_2^2 = y_2$, $x_3^2 = y_3$, $x_4 = V_1$, $x_5 = V_2$. Переопределение независимых переменных x_1, x_2, x_3 с помощью квадратичной зависимости легко решает проблему физической реализуемости без включения дополнительных ограничений в процесс оптимизации. Целевая функция $C(X)$ задается следующей формулой: $C(X) = (x_4 - k_1)^2 + (x_5 - k_2)^2$. Модель цепи (4) включает два уравнения (M=2). Функции $g_j(X)$ определяются следующими формулами:

$$g_1(X) \equiv (1 - x_4)x_1^2 - (x_4 - x_5)(y_0 + a(x_4 - x_5)^2) - x_4 x_2^2 = 0, \qquad (7)$$

$$g_2(X) \equiv (x_4 - x_5)(y_0 + a(x_4 - x_5)^2) - x_5 x_3^2 = 0.$$

Процедура оптимизации (2), (5) включает пять уравнений. Согласно методологии первого уровня обобщения структурный базис включает четыре различных стратегии проектирования с управляющим вектором: (11100), (11101), (11110), (11111). Тем не менее, в рамках методологии второго уровня обобщения полный структурный базис включает 16 различных стратегий проектирования ($\sum_{i=0}^{2} C_5^i = 16$). Система (7) решается методом Ньютона-Рафсона.

Результаты проектирования для некоторых новых стратегий, а также «старых» стратегий приведены в табл. 1.

Последние четыре стратегии этой таблицы соответствуют первому уровню обобщения, т. е. «старым» стратегиям. Необходимо отметить, что некоторые «новые» стратегии имеют процессорное время значительно меньше, чем все «старые» стратегии.

Стратегия 1, с управляющим вектором (01011), имеет минимальное процессорное время среди всех стратегий и максимальный выигрыш, равный 12,2 раза по сравнению с традиционной стратегией проектирования 8, соответствующей управляющему вектору (11100). В то же время, модифицированная традиционная стратегия проектирования, которая соответствует управляющему вектору (11111) и является наилучшей среди «старых» стратегий

проектирования, имеет выигрыш по времени только в 1,67 раза по сравнению с ТСП. Налицо дополнительный выигрыш в 7,3 раза при наличии обобщения второго уровня.

Таблица 1

N п/п	Управляющий вектор U (u1, u2, u3, u4, u5)	Число итераций	Процессорное время, с
1	(0 1 0 1 1)	5	0,000851
2	(0 1 1 1 1)	178	0,016671
3	(1 0 0 1 1)	201	0,026235
4	(1 0 1 1 1)	3162	0,300012
5	(1 1 0 0 1)	23	0,002205
6	(1 1 0 1 0)	49	0,100011
7	(1 1 0 1 1)	49	0,002405
8	(1 1 1 0 0)	107	0,010365
9	(1 1 1 0 1)	1063	0,170011
10	(1 1 1 1 0)	143	0,013115
11	(1 1 1 1 1)	243	0,006215

Следующий пример соответствует проектированию активной цепи. На рис. 2 представлена схема однокаскадного транзисторного усилителя.

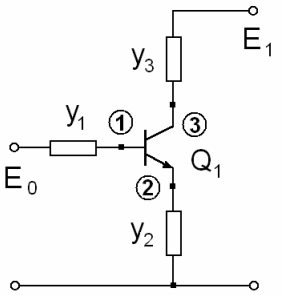


Рис. 2. Однокаскадный транзисторный усилитель

В качестве модели транзистора использовалась статическая модель Эберса-Молла, применявшаяся в системе SPICE [8]. Вектор X включает 6 компонентов: $x_1^2 = y_1$, $x_2^2 = y_2$, $x_3^2 = y_3$, $x_4 = V_1$, $x_5 = V_2$, $x_6 = V_3$. Математическая модель этой цепи (4) включает три уравнения (M=3). Процедура оптимизации включает 6 уравнений (K+M=6) и определяется системами (2), (5). Целевая функция $C(X)$ определена формулой $C(X) = [(x_5 - x_4) - m_1]^2 + [(x_6 - x_4) - m_2]^2$, где m_1, m_2 – необходимые значения напряжений на переходах транзистора. Структурный базис, соответствующий методологии первого уровня обобщения, включает 8 различных стратегий проектирования, в то время, как второй уровень обобщения позволяет увеличить полное число различных стратегий проектирования внутри структурного базиса до $\sum_{i=0}^{3} C_6^i = 42$. Стратегия, имеющая управляющий вектор (111000), является ТСП в терминах методологии первого уровня обобщения. В этом случае только три первых уравнения системы (2) включены в процедуру оптимизации, и обобщенная целевая функция $F(X, U)$ совпадает с целевой функцией $C(X)$. Модель цепи включает все три уравнения.

Стратегия 16, соответствующая управляющему вектору (111111), является МТСП. В этом случае все 6 уравнений системы (2) включены в процедуру оптимизации, но математическая модель (4) при этом исчезает. Остальные стратегии могут быть поделены на две части. Все стратегии, для которых первые три компоненты управляющего вектора равны

единице, определяют подмножество, соответствующее методологии первого уровня обобщения, т. е. «старые» стратегии. Это стратегии с номерами от 9 до 16 в табл. 2.

Таблица 2

N п/п	Управляющий вектор U (u1,u2,u3,u4,u5,u6)	Число итераций	Процессорное время, с
1	(0 1 1 1 0 0)	12850	10992,33
2	(0 1 1 1 0 1)	47	19,73
3	(0 1 1 1 1 0)	30015	10998,24
4	(1 0 1 1 1 0)	55992	25094,21
5	(1 0 1 1 1 1)	1195	170
6	(1 1 0 0 1 1)	174	60,01
7	(1 1 0 1 0 1)	606	220,21
8	(1 1 0 1 1 1)	778	139,11
9	(1 1 1 0 0 0)	9311	7977,01
10	(1 1 1 0 0 1)	7514	4989,11
11	(1 1 1 0 1 0)	75635	43053,12
12	(1 1 1 0 1 1)	324	60,11
13	(1 1 1 1 0 0)	25079	10970,12
14	(1 1 1 1 0 1)	243	40,11
15	(1 1 1 1 1 0)	10232	2398,53
16	(1 1 1 1 1 1)	2418	196,21

Видно, что две стратегии 12 и 14 имеют полное процессорное время существенно меньшее, чем другие. Стратегия 14 является оптимальной среди всех «старых» стратегий и имеет выигрыш во времени в 198 раз по сравнению с ТСП. Стратегии, пронумерованные с 1 по 8, относятся к подмножеству стратегий, которые появляются в составе «нового» расширенного базиса. Стратегия 2 имеет минимальное процессорное время среди всех исследованных и более чем двойной выигрыш во времени, чем наилучшая стратегия 14, принадлежащая «старому» структурному базису. Выигрыш по времени для стратегии 2 достигает 404 раза в этом случае.

Рассмотрим характеристики проектирования цепи, представленной на рис. 3. Этот усилитель содержит три переменные, соответствующие проводимостям схемы y_1, y_2, y_3, (K=3), и три переменные, соответствующие узловым напряжениям V_1, V_2, V_3, (M=3) в узлах 1, 2, 3. Вектор X параметров схемы содержит шесть компонент. Первые три задаются соответствующими формулами через проводимости: $x_1^2 = y_1$, $x_2^2 = y_2$, $x_3^2 = y_3$, а остальные – через узловые напряжения: $x_4 = V_1$, $x_5 = V_2$, $x_6 = V_3$.

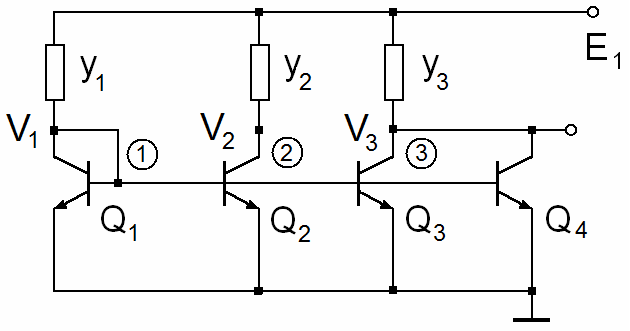


Рис. 3. Схема с четырьмя транзисторами

Управляющий вектор U включает 6 компонент $(u_1, u_2, u_3, u_4, u_5, u_6)$. Модель цепи состоит из трёх уравнений. Процедура оптимизации включает шесть уравнений. Целевая функция $C(X)$ определяется следующей формулой: $C(X) = (I_{C1} - m_1)^2$, где m_1 есть заданный коллекторный ток первого транзистора.

Структурный базис, соответствующий методологии первого уровня обобщения, включает 8 различных стратегий проектирования, в то время как второй уровень обобщения включает 42 различных стратегии.

Результаты процесса оптимизации для нескольких стратегий из старого и нового структурных базисов представлены в табл. 3.

Таблица 3

N п/п	Управляющий вектор U (u1,u2,u3,u4,u5,u6)	Число итераций	Процессорное время, с
1	(0 0 0 1 1 1)	71	0,0467
2	(0 0 1 1 1 1)	28	0,0119
3	(0 1 0 1 1 1)	25	0,0111
4	(0 1 1 1 0 1)	42	0,0176
5	(0 1 1 1 1 1)	38	0,0108
6	(1 0 1 0 1 1)	43	0,0201
7	(1 0 1 1 1 1)	49	0,0062
8	(1 1 0 1 1 1)	31	0,0051
9	(1 1 1 0 0 0)	2256	2,0992
10	(1 1 1 0 0 1)	59	0,0256
11	(1 1 1 0 1 1)	47	0,0132
12	(1 1 1 1 0 1)	34	0,0045
13	(1 1 1 1 1 1)	46	0,0036

Пять последних стратегий относятся к структурному базису первого уровня обобщения, в то время как остальные – ко второму. Для данного примера МТСП оказалась самой быстрой из всех возможных стратегий проектирования. Выигрыш по времени для этой стратегии составил 583 раза. В этом случае новый структурный базис не производит более быстрых стратегий проектирования, однако в его составе имеется большое число стратегий, имеющих выигрыш по времени более 100.

Последний пример соответствует операционному усилителю, представленному на рис. 4.

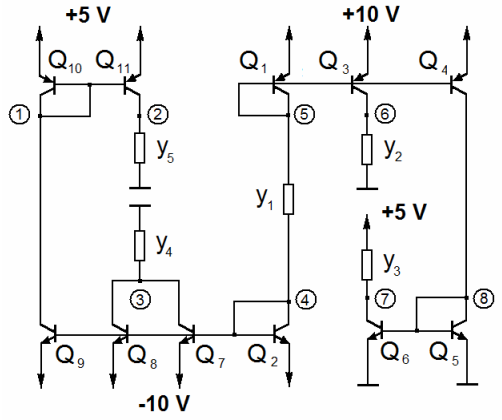


Рис. 4. Операционный усилитель

В этом случае вектор X включает 13 компонент. Пять из них определяют независимые параметры $x_1^2 = y_1$, $x_2^2 = y_2$, $x_3^2 = y_3$, $x_4^2 = y_4$, $x_5^2 = y_5$ и другие восемь компонент $x_6 = V_1$, $x_7 = V_2$, $x_8 = V_3$, $x_9 = V_4$, $x_{10} = V_5$, $x_{11} = V_6$, $x_{12} = V_7$, $x_{13} = V_8$ определяют зависимые параметры в соответствии с традиционным подходом. Целевая функция $C(X)$ определяется формулой: $C(X) = (I_{C1} - m_1)^2$, где m_1 есть заданный коллекторный ток первого транзистора.

Структурный базис состоит из 256 различных стратегий проектирования согласно методологии первого уровня обобщения. С другой стороны, структурный базис, соответствующий методологии второго уровня обобщения, состоит из $\sum_{i=0}^{8} C_{13}^i = 7099$ различных страте-

гий. Снова можно констатировать существенное расширение структурного базиса во втором случае. Результаты анализа ТСП и некоторых стратегий, имеющих процессорное время меньшее, чем ТСП, представлены в табл. 4а и 4б.

Таблица 4а

N п/п	Управляющий вектор U(u1,u2,...,u13)	Число итераций	Процессорное время, с
1	(1111100000000)	6990	24.7502
2	(1111100000001)	90	0.1454
3	(1111100000011)	246	0.3411
4	(1111100001111)	875	0.7301
5	(1111100011111)	299	0.1532
6	(1111100111111)	301	0.1213
7	(1111110000001)	159	0.2042
8	(1111110001111)	777	0.6001
9	(1111110111111)	216	0.0611
10	(1111111000001)	157	0.1453
11	(1111111011110)	59	0.0291
12	(1111111011111)	153	0.0531
13	(1111111101110)	303	0.1102
14	(1111111110111)	190	0.0751
15	(1111111111010)	132	0.0361
16	(1111111111011)	207	0.0452
17	(1111111111100)	155	0.0571
18	(1111111111101)	257	0.0573
19	(1111111111110)	121	0.0354
20	(1111111111111)	607	0.0871

Таблица 4б

N п/п	Управляющий вектор U(u1,u2,...,u13)	Число итераций	Процессорное время, с
1	(0011111011111)	131	0.0681
2	(0011111111111)	138	0.0477
3	(0101111111111)	118	0.0441
4	(0110111111111)	83	0.0343
5	(0111011111111)	142	0.0536
6	(0111101111111)	123	0.0464
7	(0111111111111)	155	0.0422
8	(1001111111111)	232	0.0754
9	(1010111111111)	338	0.0982
10	(1011101111111)	145	0.0402
11	(1011110111111)	247	0.0657
12	(1011111011111)	156	0.0478
13	(1011111101111)	502	0.1425
14	(1011111110111)	300	0.1145
15	(1011111111101)	287	0.0825
16	(1011111111110)	132	0.0425
17	(1011111111111)	77	0.0171
18	(1110111111110)	83	0.0248
19	(1110111111111)	254	0.0602
20	(1111011111111)	176	0.0339

Стратегии проектирования из табл. 4а принадлежат подмножеству стратегий методологии первого уровня обобщения. Стратегия 1, соответствующая управляющему вектору (1111100000000), является ТСП. Эта стратегия имеет наибольшее число итерационных шагов и наибольшее компьютерное время (24,75 с). Другие стратегии, представленные в этой таблице, имеют значительно меньшее число итерационных шагов и компьютерное время. Например, МТСП с управляющим вектором (1111111111111) имеет компьютерное время 0,202 с. Выигрыш во времени в этом случае составляет 123,7 раза. Стратегия 11, соответствующая управляющему вектору (1111111011110), имеет минимальное компьютерное время среди всех стратегий данного подмножества. Временной выигрыш в этом случае составляет 850 раз по сравнению с ТСП.

Стратегии из табл. 4б принадлежат подмножеству новых стратегий, появившихся в рамках методологии второго уровня обобщения. Стратегия 17 этого подмножества соответствует минимальному времени проектирования среди всех приведенных стратегий. В этом случае выигрыш по времени составляет 1447 раз по сравнению с ТСП, т.е. имеется дополнительный выигрыш в 1,7 раза по сравнению с лучшей стратегией методологии первого уровня обобщения.

Кроме того, среди «старых» стратегий четыре имеют выигрыш во времени более чем в 500 раз, и семь имеют выигрыш более 400 раз по сравнению с ТСП. С другой стороны, среди «новых» стратегий 11 стратегий имеют выигрыш 500 раз и 13 стратегий, имеющих выигрыш более 400 раз.

4. Заключение

Анализ различных примеров проектирования аналоговых цепей, как пассивных, так и активных, показывает значительный потенциал сокращения времени проектирования при использовании новых стратегий, появляющихся в рамках более обобщенной методологии. Потенциальный выигрыш во времени, появляющийся в рамках этой методологии, существенно больше, чем в ранее развитом подходе.

Следовательно, можно констатировать, что методология второго уровня обобщения содержит более перспективные стратегии проектирования, позволяющие в большей степени сократить процессорное время. Это происходит потому, что в данном случае мы имеем дело с расширенным структурным базисом, включающим значительно большее число различных стратегий проектирования, которые появляются в рамках методологии второго уровня обобщения. Таким образом, принимая во внимание полученные результаты, можно

констатировать, что методология второго уровня обобщения дает возможность существенно улучшить основные показатели построенной ранее оптимальной теории проектирования. Дальнейшее исследование может быть сфокусировано на проблеме поиска минимальной по времени стратегии проектирования путем оптимизации управляющего вектора.

Список литературы: 1. *Rizzoli V., Costanzo A., Cecchetti C.* Numerical optimization of broadband nonlinear microwave circuits // IEEE MTT-S Int. Symp. 1990. Vol. 1.P. 335-338. **2.** *Ochotta E.S., Rutenbar R.A., Carley L.R.* Synthesis of high-performance analog circuits in ASTRX/OBLX // IEEE Trans. on CAD. 1996. Vol. 15, № 3. P. 273-294. **3.** *Каширский И.С., Трохименко Я.К.* Обобщенная оптимизация электронных схем. Киев: Техника, 1979. **4.** *Zemliak A.M*, Analog system design problem formulation by optimum control theory // IEICE Trans. on Fundamentals of Electronics, Comunications and Computer Sciencies, 2001. Vol. E84-A, № 8. P. 2029-2041. **5.***Земляк А. М.* Проектирование аналоговых цепей методами теории управления. I. Теория // Изв. высш. учеб. заведений. Радиоэлектроника. 2004. Т. 47, № 5. С. 18–28. **6.***Земляк А. М.* Проектирование аналоговых цепей методами теории управления. II. Численные результаты // Изв. высш. учеб. заведений. Радиоэлектроника. 2004. Т. 47, № 6. С. 65-71. **7.** *Земляк А. М.* Проектирование аналоговой системы как управляемый динамический процесс // Нелинейный мир. 2006. № 11. С. 609–618. **8.** *Massobrio G., Antognetti P.* Semiconductor Device Modeling with SPICE. N.Y.: Mc. Graw-Hill, Inc., 1993.

Поступила в редколлегию 12.06.2012

Земляк Александр Михайлович, канд. техн. наук, доцент, доцент Физико-технического института, Национальный технический университет Украины «КПИ». Научные интересы: анализ и проектирование ВЧ и СВЧ электронных цепей, моделирование, анализ и оптимизация СВЧ приборов, оптимальное проектирование электронных систем. Адрес: Украина, 03056, Киев, пр. Перемоги, 37, корп. 11, тел. +038 044 4068104. E-mail: azemliak@mail.ru.

Маркина Татьяна Михайловна, старший преподаватель Физико-технического института, Национальный технический университет Украины «КПИ». Научные интересы: оптимальное проектирование электронных систем. Адрес: Украина, 03056, Киев, пр. Перемоги, 37, корп. 11, тел. +038 044 4068104. E-mail: martm@inbox.ru 099-744-93-75

УДК 638.562:51.65.012

И.В. ЛЕВЫКИН, Е.В. ЛОГВИНЕНКО

УСОВЕРШЕНСТВОВАННЫЕ МАТЕМАТИЧЕСКИЕ МОДЕЛИ ОПИСАНИЯ ХАРАКТЕРИСТИК ОПЕРАЦИИ ЗАКАЗА И ОБОРУДОВАНИЯ ПОЛИГРАФИЧЕСКОГО ПРЕДПРИЯТИЯ

Предлагается разработка усовершенствованных математических моделей описания характеристик операции заказа и оборудования полиграфического предприятия. Разработанные модели отражают специфические свойства печатных изданий и технологии их производства, а также параметры оборудования полиграфического предприятия. Приведенные модели могут быть использованы для реализации задач планирования и регулирования выполнения заказов и загрузки оборудования полиграфического предприятия.

1.Введение

Выполнение портфеля заказов полиграфического производства состоит в автоматизации процесса составления и редактирования календарного плана выполнения заказов, а также календарного плана загрузки оборудования производственных отделов полиграфического предприятия. Различные варианты постановки и решения задачи календарного планирования для производств дискретного типа (к которому относятся полиграфические предприятия) были предложены в работах В.С. Танаева, В.В. Шкурбы [1], Р.В. Конвейя, В.Л. Максвелла [2], Т.П. Подчасовой [3], М.Х. Прилуцкого [4]. Актуальность исследования заключается в том, что разработанные ранее модели операций и оборудования описывают только общие для дискретного типа производств характеристики этих объектов, которые не являются достаточными для осуществления оптимальных процессов планирования в рамках полиграфического производства.

Целью данного исследования является разработка моделей, параметры которых отражают особенности производства печатных изданий и позволяют использовать их в процессах планирования загрузки оборудования для оптимального выполнения заказов полиграфического предприятия.

2. Разработка математической модели описания характеристик заказа и операции полиграфического предприятия

Для разработки модели введем следующие обозначения: $Z(k)((z_1,z_2,...,z_i,...,z_{N^z(k)})$ – портфель заказов полиграфического предприятия на k – момент их поступления в отдел ПДС, где z_i – i-й заказ портфеля $Z(k)$, а $N^z(k)$ – количество заказов z_i в портфеле $Z(k)$. При этом $z_i ((O^i_1,O^i_2,...,O^i_j,...,O^i_{N^O})$, где O^i_j – j-я операция i-го заказа, а N^O – количество операций в i-м заказе портфеля $Z(k)$.

Каждый заказ z_i портфеля $Z(k)$ описывается набором параметров. Для планирования дискретного производства ключевыми являются такие параметры как время начала выполнения, директивный срок и длительность выполнения i-го заказа портфеля $Z(k)$. Обозначим через S_i^{start} дату начала выполнения, S_i^{finish} – дату окончания, S_i^{direct} – директивный срок выполнения, а i– длительность выполнения i-го заказа портфеля $Z(k)$.

Так как в условиях полиграфического производства выполнение заказа представляет собой последовательное выполнение технологических операций, то длительность его выполнения будет равна сумме длительностей выполнения всех операций данного заказа.

Поскольку технические характеристики оборудования одного цеха могут отличаться, то в зависимости от того, на каком оборудовании будет выполняться соответствующая технологическая операция, длительность ее выполнения будет изменяться. Обозначим длительность выполнения j-й операций i-го заказа портфеля $Z(k)$ на определенном оборудовании цеха $\blacklozenge^i_J(E^n_m)$, где E^n_m – оборудование m цеха n. Так как для расчета суммарной длительности выполнения заказа необходимо знать время начала и окончания выполнения операций, обозначим $t^{i\ start}_J$ – момент начала выполнения, $t^{i\ finish}_J$ – момент окончания выполнения i-й операции j-го заказа портфеля $Z(k)$.

Следует отметить, что большинство заказов полиграфического производства состоит из нескольких операций, каждая из которых имеет различное значение параметра $t^{i\ start}_J$, поэтому для первой операции заказа z_i будет выполняться равенство $t^{i\ start}_J (S^{start}_i)$.

Специфической особенностью полиграфического производства является тот факт, что при оформлении заказа помимо директивных сроков его выполнения учитывается также максимально возможное превышение этих сроков. Это объясняется характеристиками производимой продукции, а также тем, что из-за сбоев на производстве и перемещения времени выполнения отдельных технологических операций такие ситуации возникают достаточно часто. При превышении запланированных директивных сроков выполнения заказа производитель теряет часть прибыли от заказа. Сумма такой потери также оговаривается заранее. При этом планирование таких потерь позволяет прогнозировать прибыль предприятия. Специфической особенностью заказа является Ex_i – максимально возможное превышение директивных сроков выполнения i-го заказа портфеля S^{start}_i и его приоритет A_i, который представляет собой абстрактный показатель «важности» заказа. В зависимости от модели взаимоотношений с контрагентами приоритет может зависеть от суммы прибыли, получаемой от выполнения заказа, периодичности поступления заказов или сроков сотрудничества с данным контрагентом и т.п. Значение приоритета заказа влияет на планирование производства, так как в первую очередь в календарный план заносят операции заказа с наибольшим приоритетом.

Одними из ключевых параметров для планирования выполнения заказа является его красочность и тираж. От красочности заказа зависит количество необходимых печатных

форм, а также время на приладку печатной машины, а от тиража, в свою очередь, напрямую зависит время выполнения заказа. Также важен формат заказа, так как заказы малого формата могут выполняться на оборудовании большего формата, однако использование малоформатного оборудования для крупноформатных заказов невозможно. Для разработки модели заказа и операции полиграфического предприятия обозначим C_i – красочность, F_i – формат, Tp_i – тираж i-го заказа портфеля $Z(k)$. Следует также учесть, что в полиграфической отрасли разделяют формат, красочность, тираж заказа и операции, поскольку красочность и формат операции могут не совпадать с красочностью и форматом заказа в целом.

С учетом операций обозначим C_j^i – красочность, F_j^i – формат, Tp_j^i – тираж $_j$-й операции i-го заказа портфеля $Z(k)$.

Предлагается ввести параметр субподрядной операции $D_j^i \circ \overline{(0,1)}$, который принимает значение 1, если операция является субподрядной, и 0 – в противном случае.

Поскольку особенностью полиграфической отрасли Украины является наличие различного по времени эксплуатации и техническим характеристикам оборудования, важным параметром операции выступает надежность ее выполнения на данном оборудовании. Предлагается обозначить надежность выполнения j-й операции i-го заказа R_j^i, которая равна надёжности работы оборудования, на котором она выполняется.

С учетом введенных выше параметров модель описания характеристик заказа полиграфического производства представим так:

$$z_i \left((S_i^{start}, S_i^{finish}, S_i^{direct}, (_i, Tp_i, Ex_i, A_i, C_i, F_i),\right.$$

а модель описания характеристик операции заказа полиграфического производства выражением:

$$O_j^i \blacklozenge \left(\blacklozenge_j^i (E_m^n), t_J^{i\,start}, t_J^{i\,finish}, C_j^i, F_j^i, Tp_j^i, D_j^i, R_j^i \right).$$

3. Разработка усовершенствованной математической модели описания характеристик оборудования полиграфического предприятия

Выполнение портфеля заказов требует наличия необходимого количества производственного оборудования. Для разработки математической модели планирования выполнения портфеля заказов полиграфического производства необходимо его техническое оснащение в виде совокупности станков, которые объединены в цеха. Организационную структуру полиграфического предприятия представим множеством цехов $D\left((d_1, d_2, ..., d_n, ..., d_{N^D}) \right.$, где d_n – цех, а N^d – количество цехов предприятия. Цех d_n представляет собой совокупность некоторого количества оборудования – $d_n \left((E_1^n, E_2^n, ..., E_m^n, .., E_{N^{E(n)}}^n) \right.$, где E_m^n – оборудование m цеха n, а $N^{E(n)}$ – количество оборудования в n-м цеху. Каждый станок E_m^n цеха d_n описывается набором определенных параметров.

В условиях полиграфического производства оборудование может быть в одном из трех состояний – свободно, работает, находится на профилактическом ремонте. Обозначим состояние $b_m^n \measuredangle(\overline{1,3})$ станка E_m^n в момент времени k.

Таким образом, вектор состояний оборудования полиграфического предприятия можно представить в виде:

$$B_m^n(k) \blacklozenge (b_m^n(k_0), b_m^n(k_0 \blacklozenge 1), ..., b_m^n(k), .., b_m^n(k_0 \blacklozenge K)),$$

где k_0 – момент начала планового периода; K – момент его окончания.

При разработке модели выбора оборудования полиграфического предприятия следует учитывать взаимосвязь технических характеристик заказов (красочность, формат) и оборудования. Обозначим C_m^n – красочность, F_m^n – формат, V_m^n – скорость обработки одной единицы тиража, R_m^n – надежность m-й единицы оборудования цеха d_n.

Поскольку надежность работы оборудования зависит от большого количества различных факторов (продолжительность эксплуатации, наличие или отсутствие аппаратных средств контроля качества продукции), то целесообразным является ее определение с использованием метода экспертных оценок [5].

С учетом предложенных параметров модель описания характеристик оборудования полиграфического предприятия можно представить в виде:

$$E^n_m \left((B^n_m(k), R^n_m, C^n_m, F^n_m, V^n_m \right).$$

4. Выводы

Результатом исследования является разработка усовершенствованной математической модели описания характеристик операции заказа полиграфического предприятия, которая учитывает такие параметры, как красочность, формат, тираж, надежность выполнения, субподряд, что соответствует характерным особенностям производства полиграфической продукции.

Также разработана усовершенствованная модель описания характеристик оборудования полиграфического предприятия, которая, в отличие от существующих, учитывает такие характеристики печатных изданий, что позволяет планировать загрузку оборудования и выполнение заказов с учетом специфики технологических процессов полиграфической отрасли.

Литература: 1. *Танаев В.С., Шкурба В.В.* Введение в теорию расписаний. М.: Наука, 1975. 256 с. **2.** *Конвей Р.В., Максвелл В.Л., Миллер Л.В.* Теория расписаний. М.: Наука, 1975. 360 с. **3.** *Подчасова Т.П., Лагода А.П., Рудницкий В.Ф.* Управление в иерархических производственных структурах. Киев: Наук. думка, 1984. 189 с. **4.** *Прилуцкий М.Х., Вяхирев Д.В.* Многостадийные задачи альтернативного распределения ресурсов // Вестник Нижегородского государственного университета. Математическое моделирование и оптимальное управление. 2002. № 25(1). С.224-233. **5.** *Мушик Э., Мюллер Г.* Методы принятия технических решений. М.: Мир, 1990. 208 с.

Поступила в редколлегию 11.06.2012

Левыкин Игорь Викторович, канд. техн. наук, доцент кафедры медиасистем и технологий ХНУРЭ. Научные интересы: разработка автоматизированных систем управления полиграфическим предприятием, автоматизация полиграфических процессов WorkFlow. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. +38 (057) 702-13-78.

Логвиненко Екатерина Витальевна, аспирант кафедры медиасистем и технологий ХНУРЭ. Научные интересы: автоматизация полиграфических процессов WorkFlow. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. +38 (057) 702-13-78.

УДК 519.7

Т. Н. ФЕДОРОВА

О ПОДХОДЕ К ПОСТРОЕНИЮ ЦЕПОЧЕК ЛЕКСИЧЕСКИХ ЕДИНИЦ УКРАИНСКОГО ЯЗЫКА В ЛЕКСИКОГРАФИЧЕСКОЙ СИСТЕМЕ ЭЛЕКТРОННОГО ТОЛКОВОГО СЛОВАРЯ

Рассматривается дальнейшее развитие метода нахождения n-го линейного логического преобразования для построения цепочек в лексикографической системе электронных толковых словарей. Модификация метода характеризуется заданием исходной семантической зависимости на каждом этапе вычисления. Рассматривается реализация метода программой «Побудова гіперланцюгів», которая позволяет строить, редактировать и анализировать цепочки.

1. Введение

В связи с потребностями настоящего времени появляются новые разделы лексикографии, которые подают примеры дальновидных обобщений понятия словаря, а идентификация информационных процессов побуждает к ускорению разработки различных методик формализации языка, а также к созданию все более мощных методов лексикографирования

явлений предметного мира. Пути, по которым движется сегодня лексикография, во многом определяются внешними факторами, среди которых глобализация, становление индустрии знаний и вызванная этим потребность в интеллектуальных средствах экстракции знаний, способных в реальном времени обрабатывать сверхбольшие массивы естественно-речевой информации. В результате сегодня словарное дело переживает особый этап своего развития, находясь под влиянием новых общественных потребностей и новых методов обработки информации, а также используя возможности применения компьютерных технологий при описании и представлении как собственно лингвистической, так и экстралингвистической информации.

Поскольку компьютерная лексикография требует максимальной формализации своего объекта, возникла проблема в углублении содержания и формализации самого понятия словаря как специфического культурно-информационного объекта в процессе развертывания фундаментальных для языка отношений «субъект - объект » и « форма-содержание ». Ответом на эту потребность стала разработка лексикографических систем, которые поясняют внутренние механизмы, побуждающие к приобретению естественно-речевой информацией словарной формы.

Одна из проблем лексикографии касается применения словарей в формировании лингвистических компонент концептографических систем представления знаний и использовании их в средствах экстракции знаний. Это требует не только больших словарных массивов, но и побуждает к нахождению в традиционных словарных текстах скрытых семантических структур. С другой стороны, онто- и концептографическая проблематика требует более активной и содержательной интеграции лексикографии в современную индустрию знаний [1].

Целью данной работы является модификация метода линейного логического преобразования n-й степени для построения цепочек лексических единиц украинского языка[2, 3]. Это позволит повысить скорость и точность обработки словарных статей с помощью анализа отношений толкования и построения гиперцепочек между лексическими единицами украинского языка. Для достижения поставленной цели необходимо решить следующие задачи: усовершенствовать метод, построить алгоритм и программно реализовать его.

2. Метод построения цепочек лексических единиц украинского языка

Цепочки вида «толкуется через» используют для построения систем, которые могли бы находить в тексте или в его фрагменте не только конкретно заданное слово, но и это слово по его содержанию, описанию. Примерами таких систем являются «ПроСеКа» [4] и «СКАЗКА-2» [5].

В работе [2] изложен и обоснован метод нахождения n-го линейного логического преобразования. Было доказано утверждение о том, что при нахождении степени линейного логического преобразования, если на двух последующих шагах значение преобразования повторяется, то такое линейное преобразование и будет искомым. Этот же критерий нахождения n-го линейного логического преобразования был использован при решении задачи построения цепей лексических единиц.

Рассмотрим в общем виде метод нахождения n-го линейного логического преобразования. $P(x)$, $Q(y)$ – предикаты, $K(x,y)$ – ядро линейного логического преобразования, M – множество, элементы которого являются логическими векторами. Линейные логические преобразования можно представить в виде $Q(y) = \exists x \in M(K(x,y)P(x))$.

Приведем в общем виде формулу вычисления преобразования $P^{(n)}(x)$ и $Q^{(n)}(y)$ в зависимости от предикатов $P(x)$ и $Q(y)$ соответственно.

Пусть $Q(y) = K(x,y)P(x)$, $P'(x) = K(y,x)Q(y)$. Преобразование из $P'(x)$ представим в следующем виде:

$$Q'(y) = K(x,y)P'(x) \overset{(1)}{=} K(x,y)K(y,x)Q(y) = KQ(y),$$

$$P'(x) \overset{(1)}{=} K(y,x)K(x,y)P(x) = K'P(x).$$

Преобразование из $P''(x)$ представим в виде:

$$Q''(y) = K(x,y)P''(x) = K(x,y)K(y,x)Q'(y) \overset{(3)}{=} KKQ(y),$$

$$P''(x) = K(y,x)Q'(y) = K(x,y)K(x,y)P'(x) \overset{(4)}{=} K'K'P(x).$$

Действуя аналогичным способом, получаем формулу вычисления n-го линейного логического преобразования вида:

$$Q^{(n)}(y) = \bigwedge_{i=1}^{n} K_i Q(y), \text{ где } K_i = K = K(x,y)K(y,x),$$

$$P^{(n)}(x) = \bigwedge_{i=1}^{n} K_i' P(x), \text{ где } K_i' = K' = K(y,x)K(x,y).$$

Если линейные логические преобразования n-й и $n+1$-й степени совпадают, то n-е логическое преобразование далее не изменится, оно стабилизируется на n-м шаге.

Алгебра конечных предикатов позволяет формализовать подход к построению цепочек следующим образом: пусть $P(x)$ – слово, $K(x_{n-1},x_n)$ – семантическая зависимость, определяющая функцию толкования, M – множество всех слов в словарных статьях электронного толкового словаря.

Задаем $P(x_1)$, $K_1(x_1,x_2)$. Вычисляем $P(x_2)$:

$$P(x_2) = \exists x_1 \; P(x_1) \, K_1(x_1, x_2). \quad (1)$$

Задаем $K_2(x_2,x_3)$. Вычисляем $P(x_3)$:

$$P(x_3) = \exists x_2 \; P(x_3) \, K_2(x_2, x_3). \quad (2)$$

Задаем $K_3(x_3,x_4)$. Вычисляем $P(x_4)$:

$$P(x_4) = \exists x_3 \; P(x_3) \, K_3(x_3, x_4) \quad (3)$$

и т.д. Вычисление будет проходить до того момента, пока не выполнится условие завершения построения цепочки:

$$x_n = x_i \text{ если } \exists \; K_i(x_i, x_n), \; i = \overline{1, n-1}. \quad (4)$$

Приведем в общем виде формулу вычисления преобразования $P(x_n)$:

$$P(x_n) = \exists x_{n-1} \; P(x_{n-1}) \, K_{n-1}(x_{n-1}, x_n). \quad (5)$$

На рис. 1 изображено графическое представление метода построения цепей лексических единиц, где x – слово, $x(i,j)$, i – номер уровня, j – индекс слова на уровне i.

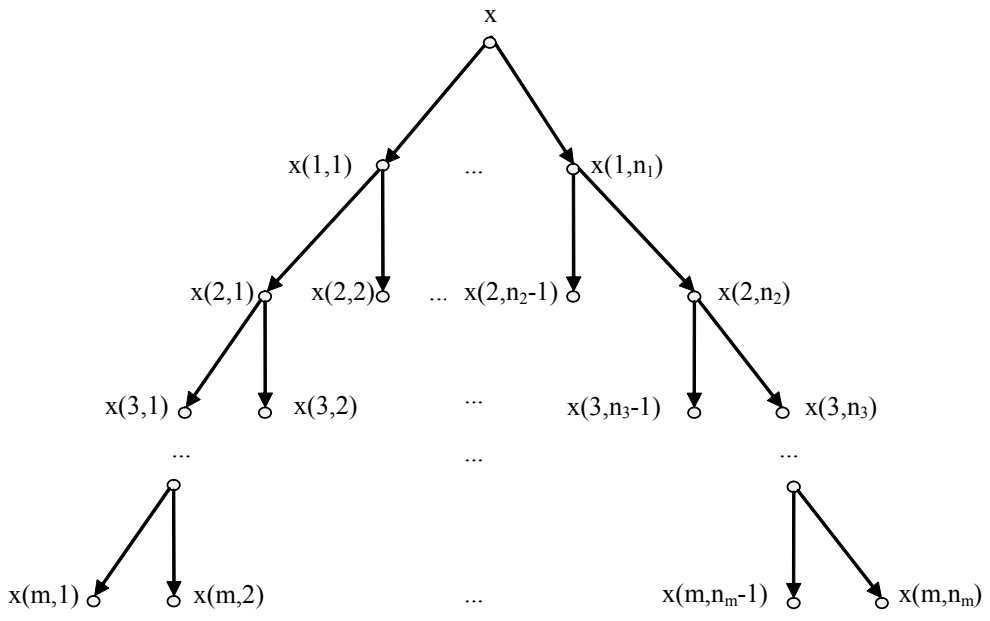


Рис. 1. Графическое представление метода

Рассмотрим пример вычисления линейного логического преобразования метода построения цепочек лексических единиц для слова «Абажур». Цепочка «Абажур -> прилад -> пристрій -> обладнання -> прилад».

«Абажур – **частина світильника**, звичайно у вигляді ковпака, **прилад** призначений для зосередження і відбиття світла та захисту очей від його впливу.»

«Прилад – 1. **Інструмент**, **предмет**, який використовується для виконання певної дії. 2. Спеціальний **пристрій**, призначений для певної мети (вимірювання чого-небудь, управління чимось, контролю, спостереження за чим-небудь і т. ін.). 3. Сукупність відповідних інструментів, предметів, необхідних для виконання певної роботи.»

«Пристрій – **пристосування**, **обладнання**, за допомогою якого виконується яка-небудь робота або спрощується, полегшується певний виробничий процес.»

«Обладнання – сукупність **механізмів**, **приладів**, необхідних для чого-небудь; **спорядження**.»

Задаем

$$P(x_1) = \{x_1^{\text{абажур}}\},$$

$$K_1((x_1^{\text{абажур}} x_2^{\text{частина}}) \vee (x_1^{\text{абажур}} x_2^{\text{світильник}}) \vee (x_1^{\text{абажур}} x_2^{\text{прилад}})),$$

вычисляем

$$P\left(x_2^{\text{прилад}}\right) = x_1^{\text{абажур}} \wedge x_1^{\text{абажур}} x_2^{\text{прилад}}.$$

Задаем

$$K_2((x_2^{\text{прилад}} x_3^{\text{інструмент}}) \vee (x_2^{\text{прилад}} x_3^{\text{предмет}}) \vee (x_2^{\text{прилад}} x_3^{\text{пристрій}})),$$

вычисляем

$$P\left(x_3^{\text{пристрій}}\right) = x_2^{\text{прилад}} \wedge x_2^{\text{прилад}} x_3^{\text{пристрій}}.$$

Задаем

$$K_3((x_3^{\text{пристрій}} x_4^{\text{пристосування}}) \vee (x_3^{\text{пристрій}} x_4^{\text{обладнання}}))$$

вычисляем

$$P\left(x_4^{\text{обладнання}}\right) = x_3^{\text{пристрій}} \wedge x_3^{\text{пристрій}} x_4^{\text{обладнання}}$$

Задаем

$$K_4((x_4^{\text{обладнання}} x_5^{\text{механізм}}) \vee (x_4^{\text{обладнання}} x_5^{\text{прилад}}) \vee (x_4^{\text{обладнання}} x_5^{\text{спорядження}}))$$

вычисляем

$$P\left(x_5^{\text{прилад}}\right) = x_4^{\text{обладнання}} \wedge x_4^{\text{обладнання}} x_5^{\text{прилад}}$$

На 5-м шаге выполнится условие завершения построения цепочки:

$$x_5^{\text{прилад}} = x_2^{\text{прилад}} \text{ для } \exists K_2(x_2^{\text{прилад}} x_5^{\text{прилад}}).$$

3. Программная реализация метода построения цепочек лексических единиц

На основе метода построения цепочек лексических единиц для украинского языка разработана программа «Побудова гиперланцюгів», которая предназначена для описания семантических отношений между лексическими единицами естественного языка.

Для построения гиперцепей, которые связываются отношением «толкуется через», используется толковый словарь «Виртуальной лексикографической лаборатории Украинского языково-информационного фонда» [3]. Целью системы является автоматизация обработки текстов с помощью анализа отношений и построения цепочек между лексическими единицами украинского языка. С помощью программы возможно строить, редактировать и анализировать цепочки.

Программа позволяет выбирать путь проведения поиска по базе гиперцепочек или по электронному толковому словарю украинского языка. Программа включает две базы, в одной хранятся цепочки, разработанные вручную, а в другой - цепочки, которые строятся на базе электронного толкового словаря. Если для слова уже была построена цепочка с помощью электронного словаря, программа будет выводить результат, ранее сохраненный

в базе, что позволяет сократить время их построения. Если же пользователю необходимо построить гиперцепочку заново, то задается параметр «принудительный поиск», и программа будет строить заново цепочку, обращаясь только к электронному словарю. Есть возможность одновременного поиска по двум базам слов. Результат выводится в отдельных окнах так, что пользователь может сравнивать полученные результаты (рис. 2).

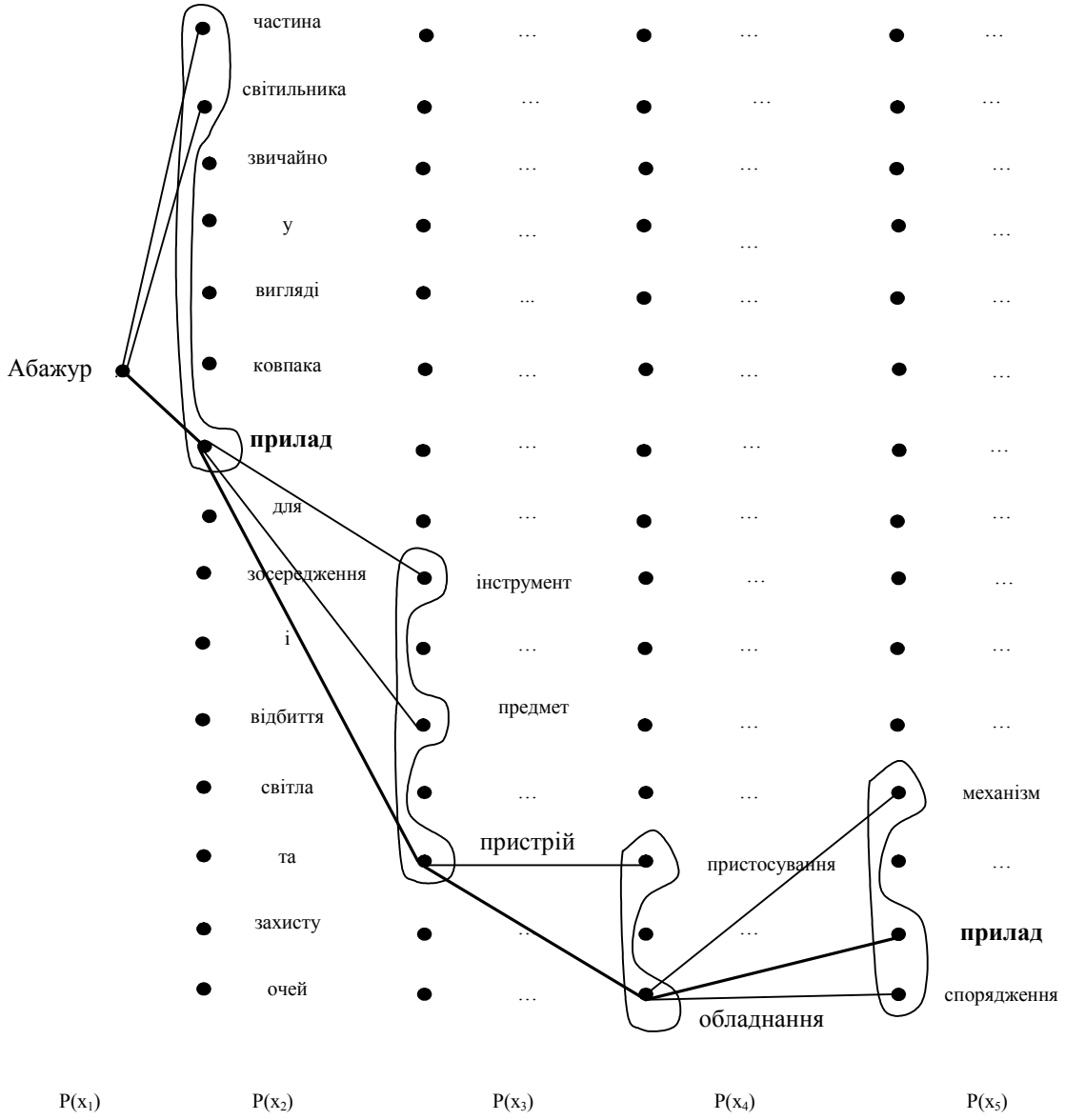


Рис. 2. Графическое представление одного из вариантов построения цепочек для слова «Абажур»

При построении гиперцепочки слова и по словарю, и по базе выводится комбинированный результат (рис. 3).

Программа позволяет контролировать каждый шаг ее выполнения, для этого необходимо установить отметку возле поля «Подтверждать каждый шаг».

Каждое слово в гиперцепочке можно редактировать следующим образом: добавить или удалить слово, для каждого слова отдельно можно отобразить дочерние слова (рис. 4). Для вызова окон редактирования слов нужно кликнуть на нужном слове и в контекстном меню выбрать вид редактирования (рис. 5).

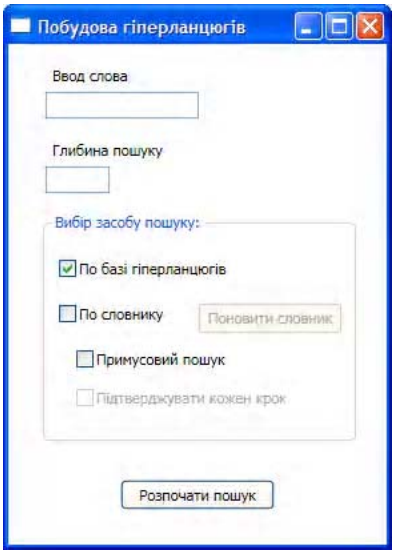
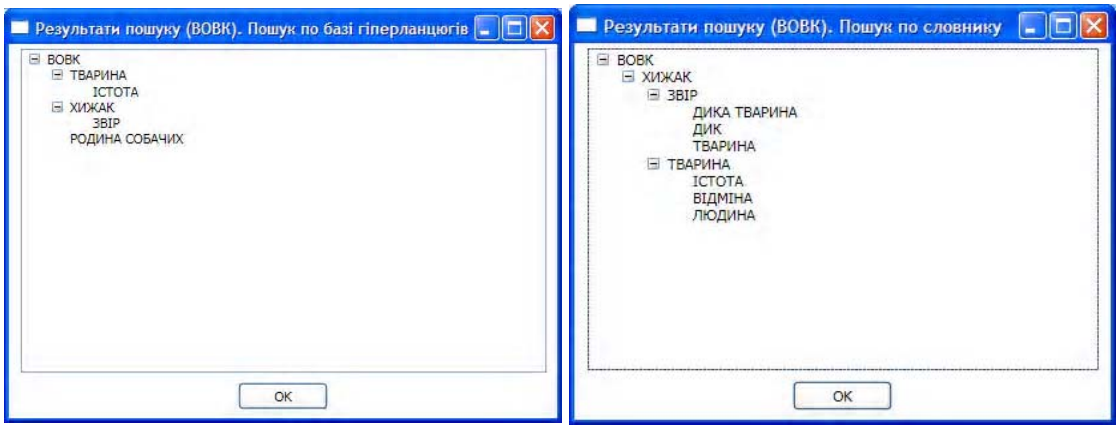


Рис. 3. Интерфейс программы «Побудова гіперланцюгів»



а б

Рис. 4. Результаты построения гиперцепочки для слова «вовк»:

а – по базе гиперцепочек; б – по словарю

Алгоритм, выполняемый для автоматизации поиска слов в электронном словаре:

1. Поиск словарной статьи. Слово, для которого выполняется поиск, вводится в поле «Найти слово» электронного словаря. Вызывается словарная статья путем вызова действия кнопки «Найти».

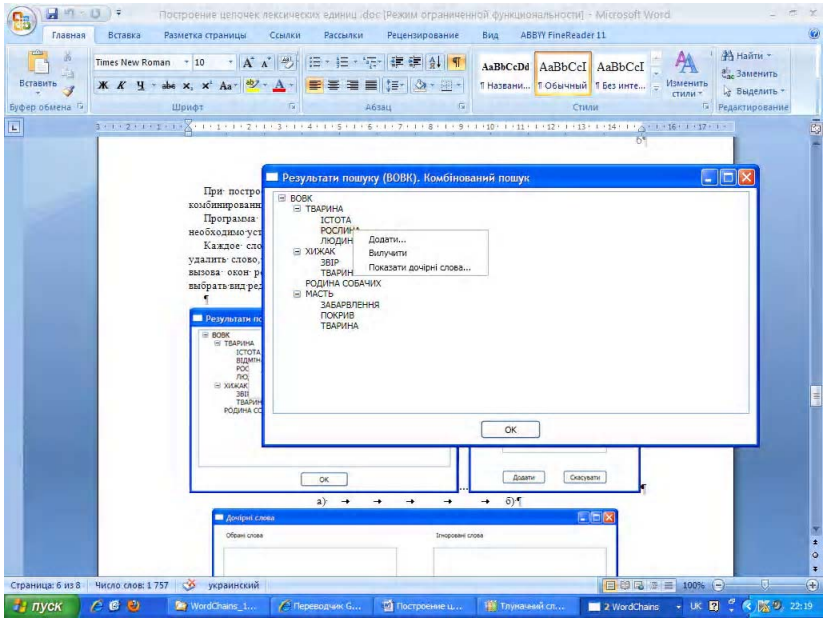
2. Анализ результата поиска. Из полученной статьи выделяются слова, дополнительные свойства слова, является ли слово существительным, и выделяется сама словарная статья, которая разделяется на отдельные слова.

3. Сбор информации о каждом слове словарной статьи. Проверяется, есть ли найденное слово в локальной базе данных. В случае, если слово отсутствует, вызывается новая соответствующая статья.

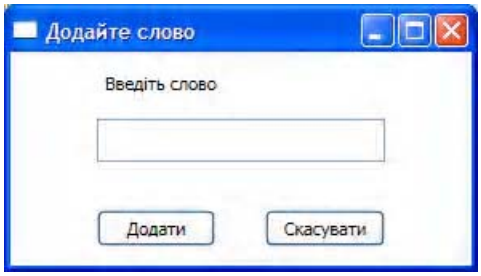
4. В новой статье выбираются слова. В случае если выбранное слово является существительным – оно добавляется в локальную базу данных, если нет – в список игнорируемых слов.

5. Вызывается кнопка «Назад» для возврата к основной статье.

6. Повторяется пункт 3 для каждого слова статьи.



а



б

в

Рис. 5. Меню для вызова окна редактирования слова (а); окно добавления слов (б); окно редактирования дочерних слов (в)

4. Выводы

Научная новизна: получил дальнейшее развитие метод нахождения n-го линейного логического преобразования для построения цепочек в лексикографической системе электронных толковых словарей путем задания исходной семантической зависимости на каждом этапе вычисления. Также рассмотрена реализация метода программой «Побудова гіперланцюгів», которая позволяет строить, редактировать и анализировать цепочки.

Практическое значение: метод позволяет распараллелить процесс обработки словарных статей с помощью анализа отношений толкования и построения гиперцепей между лексическими единицами украинского языка.

Перспективы исследования: нахождение в традиционных словарных текстах скрытых семантических структур, что позволит создавать более мощные методы лексикографирования явлений предметного мира.

Список литературы: 1. *Широков В. А.* Комп'ютерна лексикографія. К.: Наук. думка, 2011. 351 с. **2.** *Вечирская И.Д.* Линейные логические преобразования и их применение в исскуственном интеллекте: Автореферат дисс. канд. техн. наук. Х., 2007. 28с. **3.** *Широков В.А.* Лінгвістичні та технологічні основи тлумачної лексикографії. К.: Довіра, 2010. 295 с. **4.** *Рафаева А. В.* Программа семантической классификации лексики «ПроСеКа» // Материалы международной научной конференции "Горизонты прикладной лингвистики и лингвистических технологий" (MegaLing'2009). 20-27 сентября 2009, Украина, Киев.С. 67. **5.** *Рафаева А.В.* Использование программы ПРОСЕКА в исследовании сказок // Прикладна лінгвістика та лінгвістичні технології : MegaLing-2009 : Зб. наук. пр. / НАН України. Укр. мовн.-інформ. фонд, Таврійський нац. ун-т ім. В.І.Вернадського; За ред. В.А.Широкова. К. : Довіра, 2010. С. 378–382.

Поступила в редколлегию 12.06.2012

Фёдорова Татьяна Николаевна, аспирантка кафедры програмной инженерии ХНУРЭ. Научные интересы: математическая и прикладная лингвистика, алгебра логики. Интересы: изучение иностранных языков, спортивные бальные танцы, балет, вышивка, конный спорт, катание на коньках. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 702-14-77, E-mail: tanja_fyodorova@mail.ru.

УДК 65.011.56

В.А. ФИЛАТОВ, Р.В. АРТЮХ

МОДЕЛЬ ПРЕДСТАВЛЕНИЯ ВАРИАНТОВ ТЕХНОЛОГИЧЕСКИХ ПРОЦЕССОВ В БАЗЕ ПРЕЦЕДЕНТОВ

Предлагаются структурные модели представления технологических процессов для формирования архива аналогов технологических решений. На основе технологической операции определяется принцип унифицированной детали и группового технологического процесса обработки, что дает возможность использовать способ компактного, информативного и наглядного представления структуры пооперационного технологического процесса. Усовершенствуется модель представления вариантов технологических процессов в базе прецедентов, что позволяет снизить затраты времени на поиск информации для принятия решений.

Введение

Для предприятий, производящих товары широкого потребления или продукцию по контрактам, основным элементом инновационной программы является план стратегии развития, в основе которого лежат задачи модернизации выпускаемой продукции или разработки новых образцов. Реализация этих программ в основном выполняется в сфере производства, и успех определяется уровнем технологического процесса.

Таким образом, разработка стратегии развития, связанной с освоением новых видов продукции в основном базируется на оценке технологической реализуемости планов в требуемые сроки. При оценке технологической реализуемости планов развития предприятия необходима информация, которая формируется из анализа технологической документации базовых образцов прошлых разработок.

Одним из путей сокращения времени на разработку технологических процессов (ТП) является использование предыдущего опыта разработок. Информация о накопленном опы-

те должна формироваться и храниться в компактном виде, пригодном для оперативного анализа, моделирования и принятия управленческих решений. Поэтому вопросы разработки моделей для формирования и хранения информации в архиве предприятия являются *актуальными*.

Постановка задачи

Основной целью данных исследований является формирование модели компактного и унифицированного представления технологических процессов в архиве предприятия, содержащего информацию об опыте прошлых разработок в виде технологических решений.

При формировании стратегии технологического развития производства используется архив прошлых разработок, в частности, технологических решений, в котором хранится информация, необходимая для принятия решений на предпроектной стадии планирования развития предприятия. Эту информацию можно получить из анализа документации прошлых разработок, в частности, для отобранного базового образца, наиболее подходящего по параметрам к планируемой продукции. Поиск и отбор базовых образцов-аналогов производится в системе прецедентного типа с учетом степени сходства.

Одной из основных составляющих процесса принятия решения с использованием аналогов является сбор и оценка информации. В первую очередь автоматизированной обработке подвергается формализуемая информация, имеющая обычно табличный характер. При этом зачастую остается вне поля деятельности средств обработки трудно формализуемая информация [1]. На данный момент существует достаточно большое количество различных моделей, схем и методов рассуждения на основе аналогий [2 - 4]. Анализ этих методов позволяет сделать вывод, что для решения задач настоящего исследования наиболее предпочтительными являются методы прецедентов [5, 6], позволяющие решать задачи поддержки принятия решений в сложных слабоструктурированных системах. Выбор данного подхода также обусловлен тем, что зачастую на производственном предприятии к моменту возникновения проблемы выпуска новой продукции или ее модернизации уже накоплен значительный опыт решения похожих проблем, возникавших ранее [7].

Для удобства использования опыта прошлых разработок информация первого этапа предпроектных исследований должна формироваться и храниться в компактном виде в объеме и номенклатуре, требуемых для оперативного анализа, моделирования и принятия управленческих решений.

Поскольку технологическая операция является основной структурной единицей технологического процесса и на ее основе строится принцип унифицированной детали и группового техпроцесса обработки, следует использовать способ компактного, информативного и наглядного представления структуры пооперационного технологического процесса и требуемых ресурсов для его реализации.

Разнообразие технологических процессов производства существенно затрудняет их исследование, сравнение и оценку. Из-за множества параметров, присущих различным технологическим процессам, сложно формулировать обобщенные критерии оценки, так как значимость их различна для сравниваемых вариантов. Это обстоятельство существенно усложняет способы представления технологических процессов в архиве прецедентов, особенно структурной части ТП. Для задач предпроектного анализа структуры и ресурсных параметров технологических процессов необходимо иметь возможность формализовать структуры ТП по набору обобщенных унифицированных операций, которые несут информацию по таким параметрам операций как: трудоемкость, требуемое оборудование и оснастка. Эта информация необходима для оценки реализуемости плана развития предприятия по оснащенности (возможной дооснащенности), составу работников по видам работ.

Поэтому в данной работе решаются следующие *задачи*:

1. Классификация и определение параметров основных технологических операций.
2. Построение минимальной структурной модели обобщенных технологических операций.
3. Разработка структурной модели технологического процесса на основе унифицированного представления технологической операции.

Решение задач

Для определения перечня потенциальных вариантов работ (технологических операций) необходимо описать основные ТП данного производства в терминах понятий существующих категорий. Объекты понятий, описывающие ТП, классифицируются по имеющимся категориям, описываются связи между ними и затем производится поиск ТП в базе технологических решений, близких к описываемой.

В соответствии с положениями теории прецедентов, известной как «Case-Based Reasoning» (CBR – метод рассуждений на основе прецедентов), прецедент представляет собой информационный блок, включающий в себя базовую ситуацию, соответствующее ей решение, а также перечень непосредственных исполнителей [8]. В процессе профессиональной деятельности в некоторой области формируются проблемно-ориентированные прецеденты, которые накапливаются в хранилище, в качестве которого могут выступать традиционные базы данных (БД), специализированные серверы знаний, многомерные БД, архивы и т.д. Ситуация, для которой сформирован прецедент, в дальнейшем считается опорной или базовой.

Первым этапом проектирования технологического процесса является разработка его структуры. Проработка состава и последовательности выполнения технологических операций позволяет определить перечень требуемой оснащенности и оценить объем ресурсов для дооснащения.

При определении перечня потенциальных вариантов работ (технологических операций) необходимо описать основные ТП данного производства.

Поскольку технологическая операция является основной структурной единицей технологического процесса и на ее основе строится принцип унифицированной детали и группового ТП обработки, следует использовать способ компактного, информативного и наглядного представления структуры пооперационного технологического процесса.

Анализ технологических процессов различных производств показывает, что операциями обработки, сборки, разработки (распределения), нарезки (штамповки), контроля (распределения) и испытания практически исчерпывается весь их набор [9]. Для формирования унифицированной структурной модели технологической операции введем три следующих характеристических параметра: количество входов - $n_{вх}$, количество выходов - $n_{вых}$ и учетный коэффициент передачи технологической операции K_y .

Учетным коэффициентом передачи по i-му входу и j-му выходу K_y^{ij} будем называть отношение счетного количества физических единиц материалов, комплектующих изделий, сборочных узлов и т.п. j-го выхода технологической операции $y_{выхj}$ к счетному количеству поступивших на вход технологической операции физических единиц материалов, сборочных узлов и изделий $y_{вхi}$.

Тогда можно дать следующее описание технологических операций и структуры ТП в целом.

Обработка - операция, имеющая для обрабатываемого изделия один вход $n_{вх} = 1$, один выход $n_{вых} = 1$, соответственно учетный коэффициент передачи $K_y^{ij} = 1$. Цель операции – выполнение над изделиями какой-либо технологической процедуры обработки (изменение физических или геометрических параметров изделия).

Сборка – операция, имеет несколько входов $n_{вх} = N$ и один выход $n_{вых} = 1$ с учетным коэффициентом передачи $K_y^{ij} < 1$. Цель операции – агрегирование изделий (изготовление сборочных узлов из деталей и т.п.).

Разборка (распределение) – операция, имеющая по обрабатываемому изделию один вход $n_{вх} = 1$ и несколько выходов $n_{вых} \geq 2$ с учетным коэффициентом передачи по любому выходу $K_y^{ij} > 1$.Цель операции – разагрегирование сборочных узлов, распределение комплекта одинаковых деталей на несколько потоков (разбраковка).

Нарезка (штамповка) – операция, имеющая один вход $n_{вх} = 1$ и несколько выходов $n_{вых} \geq 1$ с учетным коэффициентом передачи $K_y^{ij} \geq 1$. Цель операции – переход от групповой технологии обработки изделий к единичной.

Контроль – операция, имеющая один вход $n_{вх} = 1$ и несколько выходов $n_{вых} \geq 2$ с учетным коэффициентом передачи $K_y^{ij} > 1$ по любому выходу. Цель операции – проверка качества изделий, направленная, как правило, на сортировку (т.е. распределение по группам) последних. Для контрольной операции учетный коэффициент передачи по i-му входу K_y^{ij} является случайной величиной.

Тренировка – операция, имеющая один вход $n_{вх} = 1$ и один выход $n_{вых} = 1$, учетный коэффициент передачи $K_y^{ij} = 1$. Цель операции – улучшение качества изделий.

Испытание – операция, имеющая один вход и один выход с учетным коэффициентом передачи $K_y^{ij} = 1$. Цель операции – проверка качества изделия.

Для рассмотренных выше операций можно составить таблицу.

Таблица позволяет провести анализ структур операций. Из нее, в частности, следует:

1) операции «обработка», «тренировка», «испытание», «нарезка», «разборка» являются частными случаями по отношению к операциям «сборка» и «контроль», поэтому из дальнейшего рассмотрения могут быть исключены;

2) минимальное число входов в структуре операций «сборка» и выходов в структуре операций «контроль» не могут быть меньше двух.

Характеристические параметры технологических операций

Операция	Количество входов		Количество выходов		Учетный коэффициент передачи
	максимальное	минимальное	максимальное	минимальное	
Обработка	1	1	1	1	1
Тренировка	1	1	1	1	1
Испытание	1	1	1	1	1
Нарезка	1	1	1	1	>1
Сборка	N	2	1	1	<1
Разборка	1	1	N	2	>1
Контроль	1	1	N	2	>1

Таким образом, набор минимальных унифицированных структур технологических операций может быть ограничен двумя операциями: А и Б (рис. 1).

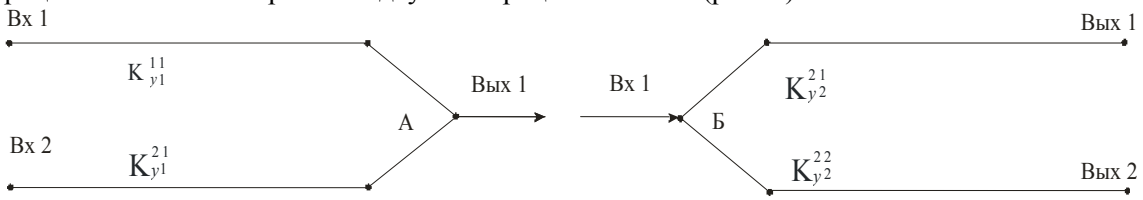


Рис. 1. Минимальная структурная модель обобщенных технологических операций:
А – сборка; Б –контроль

Поскольку любая технологическая операция на выходе кроме основной годной продукции может иметь брак, отходы и прочее, т.е. иметь несколько выходов, выполнение которых может производиться только при наличии в составе операции автоматического или ручного контроля, в ряде случаев имеет смысл объединить операции А и Б в одну унифицированную операцию, минимальная структура которой приведена на рис. 2.

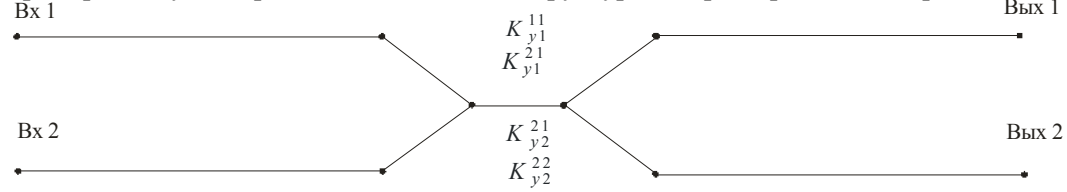


Рис. 2. Минимальная структурная модель обобщенной технологической операции
(без учета ресурсов)

Для синтеза структур технологических процессов производства используем приведенные выше модели.

На основе минимальной структуры унифицированной операции могут быть образованы структуры любых, более сложных или более простых по структуре технологических операций. Например, операции «обработка», «нарезка», «тренировка», «контроль» могут быть представлены унифицированной операцией, у которой задействованы один вход и один выход. Структуры более сложных операций компонуются на базе структуры унифицированной технологической операции путем последовательного соединения входов и выходов минимальных структур (рис. 3).

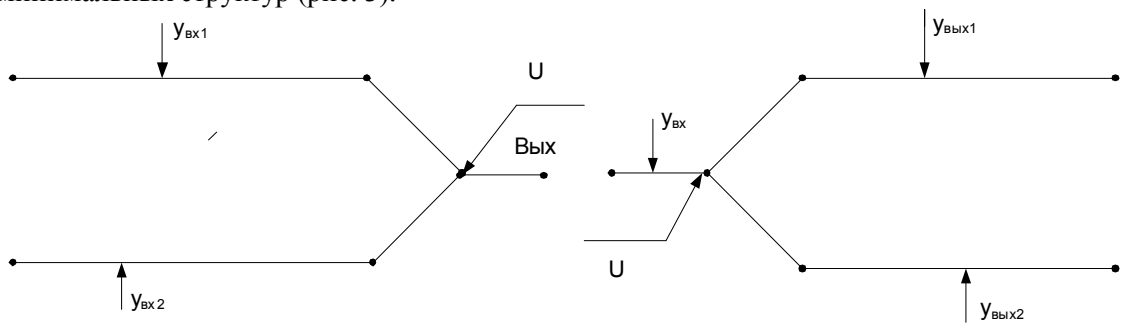


Рис. 3. Структура технологической операции, сформированной на базе унифицированной структуры

Иногда структуры технологических операций, сформированные на базе унифицированной структуры, могут быть излишне избыточными. В этом случае можно использовать специальные (для данного технологического процесса) структуры, имеющие число входов и выходов более двух.

Унифицированная структурная модель операции отражает структуру процесса функционирования производства и не позволяет определить возможности управления и необходимые ресурсы. Это объясняется тем, что в модели операции не отражены управляющие воздействия. Основными из них являются:

– воздействия по управлению производительностью операции (организация труда и обеспечение оборудованием);

– управление технологическими параметрами операции.

С учетом этих воздействий математическую модель технологической операции можно представить в виде

$$y_{вых} = f(y_{вх}, u, \theta, \eta, \xi), \quad (1)$$

где $y_{вых}$ – вектор состояния выхода операции; $y_{вх}$ – вектор состояния входа операции; u – вектор состояния ресурсов операции; θ – вектор состояния технологических параметров η; ξ – возмущающее воздействие.

На этапе предпроектного анализа вектор состояния технологических параметров θ в выражении (1) может не учитываться. Аналогично можно поступить с возмущающими воздействиями. Тогда (1) примет вид:

$$y_{вых} = f(y_{вх}, u). \quad (2)$$

В выражении (2) u – вектор состояния ресурсов, несет в себе информацию о типе привлекаемого оборудования и оснастки, трудоемкости и связанных с этим финансовых ресурсах.

Представление структурной модели технологического процесса на основе унифицированной технологической операции позволяет производить анализ технологического процесса на предпроектном этапе без необходимости разрабатывать параметрическую составляющую модели, что значительно сокращает сроки проведения оценки реализуемости планов развития.

Разработанная модель представления технологических процессов реализована в информационной подсистеме “Технология” в составе прецедентной системы принятия решений (ПСПР) для автоматизации процедур поиска, хранения и обработки информации, необходимой для принятия управленческих решений на предпроектном этапе планирования освоения новой конкурентоспособной продукции.

Основные задачи подсистемы: хранение полного систематизированного набора технической документации; осуществление информационного поиска технологической документации; вывод информации о технологических решениях по заданному набору характеристик; сравнение характеристик технологических процессов с заданным "эталоном" и ранжирование на основе процедур многокритериальной оптимизации.

Информационная поддержка подсистемы «Технология» обеспечивается работой блока «Аналог». Важным компонентом этого блока для реализации выбора вариантов развития производства является технологическая база данных, состоящая из отдельных специализированных баз данных и включающая архив прецедентов технологических решений в ПСПР. При выборе ТП-аналога на предпроектной стадии реализуется структурный поиск.

В системе поддержки прецедентных решений, построенной на основе представленного подхода к формированию базы прецедентов, временные затраты на поиск решения существенно снижаются, в результате чего быстродействие системы увеличивается.

Выводы

Предложено на предпроектном этапе формирование вариантов технологических процессов и оценку технологической реализуемости планов развития производить на основе опыта прошлых разработок, используя аппарат общей теории прецедентов с учетом специфики области применения и способов описания технологических процессов в архиве аналогов. С этой целью разработана минимальная структурная модель обобщенной технологической операции. Таким образом, *усовершенствована модель* представления вариантов технологических процессов в базе прецедентов, которая в отличие от существующих основана на унифицированных структурных моделях технологических операций, что позволяет снизить затраты времени на поиск информации для принятия решений.

Практическая значимость результатов состоит в том, что появляется возможность формирования множества вариантов для оптимизации процесса производства на основе использования опыта прошлых разработок из архива аналогов в структуре прецедентной системы. При этом необходимая для анализа информация группируется вокруг унифицированной структуры модели операции, что значительно упрощает организацию хранения и доступа к архиву прецедентов. Представление процесса изготовления типовой детали, построенной на основе унифицированных структурных моделей, позволяет сократить объем хранящейся информации и процедуры ее анализа.

Дальнейшие исследования предполагается производить в направлении использования предложенных структурных моделей унифицированных операций для построения потоковых моделей технологических процессов, анализа и вычисления параметров функционирования производственного процесса с учетом характеристик материальных потоков.

Список литературы: 1. *Бурков Ю.* Информационная система для корпораций [текст] / Ю. Бурков // Корпоративные системы. 2001. № 1. С. 20-26. **2.** *Некрасов А. Б.* Метод кластеризации и оценки множества аналогов проектных решений [текст] / А. Б. Некрасов, Н. А. Соколова, Д. Э. Лысенко // Збірник наукових праць Харківського університету Повітряних Сил. Харків: ХУПС, 2008. Вип. 2(17). С. 141-145. **3.** *Джонс М. Т.* Программирование искусственного интеллекта в приложениях [текст] / М. Т. Джонс. М.: ДМК Пресс, 2004. 312 с. **4.** *Варшавский П. Р.* Реализация схемы рассуждения на основе аналогий с помощью структурного отображения [текст] / П.Р. Варшавский // Радиотехника, электроника и энергетика: тез. докл. десятой междунар. науч.-техн. конф. студентов и аспирантов. В 3-х т., т. 1. М.: изд. МЭИ, 2004. С. 313-314. **5.** *Шерстюк В.Г.* Формальная модель гибридной сценарно-прецедентной СППР [текст] / В.Г. Шерстюк // Автоматика. Автоматизация. Электротехнические комплексы и системы. 2004. Вып. 1. С.114-122. **6.** *Павлов А.И.* Компонентный подход: модуль правдоподобного вывода по прецедентам [текст] / А.И. Павлов, А.Ю. Юрин // Программные продукты и системы. 2008. № 3. С. 55-58. **7.** *Андрейчиков, А.В.* Интеллектуальный метод синтеза технологических инноваций [текст] / А.В. Андрейчиков // Изв. вузов. Машиностроение. 2003. №10. С. 47-62. **8.** *Bergmann R.* Developing industrial case-base reasoning applications: the INRECA methodology [текст] / R. Bergmann, S. Breen, M. Goker, etc. // Lecture notes in artificial intelligence, LNAI-1612. Berlin: Springer-Verlag. 1999. P. 123-133. **9.** *Артюх Р.В.* Структурные модели технологических операций и процессов [текст] / Р.В. Артюх, А.А. Белоцкий // Вісник Херсонського національного технічного університету. 2011. № 4(43). С. 124-127.

Поступила в редколлегию 19.06.2012

Филатов Валентин Александрович, д-р техн. наук, профессор кафедры искусственного интеллекта ХНУРЭ. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 702-14-32, e-mail: Filatov_val@ukr.net

Артюх Роман Владимирович, мл. науч. сотр. НДЛ СТ ХНУРЭ. Научные интересы: стратегии развития предприятий, технологические процессы производства, теория принятия решений, многокритериальные модели. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 702-14-32, e-mail: roman_artyuh@mail.ru.

УДК 519-866

О.А. КОЖУХІВСЬКА

МОДЕЛЮВАННЯ ФІНАНСОВИХ РИЗИКІВ З ВИКОРИСТАННЯМ ЙМОВІРНІСНОГО ПІДХОДУ

Описуються основні типи ризиків у страхуванні та визначаються форми їх можливого математичного опису. Розглядається можливість застосування байєсівського підходу до побудови математичних моделей актуарних ризиків. Будується модель байєсівського типу для випадкового фінансового процесу.

1. Вступ

Повторні фінансові кризи, військова активність деяких держав і боротьба з тероризмом протягом останніх двох десятиліть свідчать про високу актуальність розв'язання задач аналізу і менеджменту фінансових ризиків. Разом з тим стає очевидним факт, що існуючі методи аналізу і моделювання ситуацій, які спрямовуються на опис та менеджмент ризиків, досить часто є недостатньо адекватними для отримання високоякісних прогнозів ймовірностей виникнення таких ситуацій та можливих втрат. Це пояснюється не стільки недостатньою увагою фахівців до таких проблем, скільки високою динамікою протікання фінансових процесів, їх надзвичайно високою розмірністю, наявністю складних ієрархічних та горизонтальних взаємозв'язків між фінансово-економічними процесами на рівні окремих фірм, галузей промисловості, макроекономіки в цілому і на глобальному рівні [1]. Іноді наявні математичні моделі є досить складними для практичного використання, а тому виникають спрощені варіанти, адекватність яких може суттєво відрізнятись від ідеалізованих. У будь-якому випадку модель – це спрощене представлення ситуацій і процесів, яке може призводити до неповноти опису та некоректних прогнозів і рішень, що на них ґрунтуються. Тому однією з першочергових задач, які виникають у менеджменті фінансових ризиків, є створення математичних моделей достатньо високого, прийнятного для практичного використання, ступеня адекватності. Моделі повинні бути зрозумілими практикам і, за необхідності, доповнені ними неврахованими елементами структури, коректними апріорними експертними оцінками окремих параметрів і початкових умов, додатковими експериментальними (статистичними) даними, можливими напрямами застосування.

Значну роль у коректному та своєчасному розв'язанні задач менеджменту фінансових ризиків відіграє системний підхід, який полягає, у даному випадку, у врахуванні впливу поточних ринкових факторів; виявленні нових факторів впливу (у тому числі прихованих); визначенні рівнів та частот впливу зовнішніх і внутрішніх факторів на протікання досліджуваних процесів; виявленні і врахуванні можливих невизначеностей структурного, статистичного і параметричного характеру; коректній постановці і розв'язанні оптимізаційних задач у тих випадках, коли це необхідно; і, наскільки це можливо, у застосуванні аналітичних критеріїв якості на всіх етапах аналізу даних [2, 3].

Аналіз протікання і взаємодії фінансових процесів протягом двох останніх десятиліть свідчить, що особливої уваги з точки зору менеджменту фінансових ризиків потребують банківська система, галузь страхування, а також великі і середні фірми та корпорації. Особливо складними і високо динамічними є фінансові процеси у галузі страхування та відповідні їм ризикові ситуації, оскільки страхування безпосередньо і тісно пов'язане з іншими локальними та глобальними процесами у міжнародній банківській системі, виробництві, туризмі, вантажному і пасажирському транспорті, з природними та індустріальними катастрофами. Саме галузь страхування, яка покликана щоденно розв'язувати складні фінансові задачі на всіх рівнях економіки та приватної діяльності, потребує високоякісних математичних моделей і методів аналізу даних і знань, безпосередньо спрямованих на практичне застосування. Математичні методи і моделі даних та методи підтримки прийняття рішень не замінюють професіоналів, які приймають остаточні рішення, але вони допомагають отримати поглиблене розуміння відповідних процесів, покращити обробку даних і експертних оцінок та прискорити процедури генерування альтернативних рішень і об'єктивно вибрати краще з них.

У даній роботі досліджуються деякі аспекти моделювання фінансових ризиків у страхуванні, а також виявляються кращі математичні моделі ризиків для практичного застосування.

Мета роботи: встановити типи ризиків у страхуванні та визначити форми їх можливого математичного опису; розглянути можливість застосування байєсівського підходу до побудови математичних моделей актуарних ризиків; побудувати модель байєсівського типу для випадкового фінансового процесу.

2. Ризики у страхуванні

Означення ризику пов'язують із ймовірністю настання подій, які супроводжуються (матеріальними) втратами, а також рівнем можливих втрат. Зокрема, Міжнародна організація із стандартів дає таке формулювання: „ризик – це комбінація ймовірності події та її наслідків " [4], а у роботі [5] дано подібне означення: „ризик – це множина сценаріїв s_i, кожний із яких характеризується ймовірністю настання p_i і наслідком c_i ". Можна сказати, що це узагальнені робастні означення, які придатні для використання при розв'язанні інженерних задач технічного чи фінансового характеру. У більшості практичних задач ризик багатовимірний, тобто існують внутрішні і зовнішні фактори ризику, які формують загальну ситуацію, пов'язану із виникненням ризиків.

Галузь страхування характеризується множиною різнотипних ризиків, зокрема, такими: 1 – індивідуальні ризики; 2 – колективні ризики для короткого (одного) періоду; 3 – колективні ризики на довгих періодах; 4 – великі розподілені ризики втрат; 5 – операційний ризик; 6 – ризик неповернення кредиту; 7 – ризик банкротства та інші [6, 7, 8]. Так, значні втрати виникають сьогодні внаслідок виникнення операційного ризику. Його можна визначити як ризик прямих або непрямих втрат, які виникають внаслідок неналежної організації виконання необхідних робіт або неналежного виконання внутрішніх процесів у фірмі, некоректної поведінки працівників та/або некоректного функціонування систем технічного забезпечення, або внаслідок впливу зовнішніх факторів. До виникнення операційного ризику призводить також відсутність належних методів та засобів менеджменту цього ж типу ризику. Так само як і інші види ризику, операційний ризик необхідно представити якісно і кількісно, для чого потрібно зібрати відповідні статистичні дані та експертні оцінки. Оскільки на виникнення цього ризику впливають найрізноманітніщі події, то задачі збору даних і оцінювання ступеня й прогнозування ймовірності такого ризику потребують значних зусиль фахівців з інформаційних технологій, математичного моделювання, прогнозування та підтримки прийняття рішень.

Можливим джерелом статистичних даних можуть бути страхові поліси, які містять інформацію стосовно страхових ризиків, що виникають внаслідок виникнення подій, пов'язаних з операційними втратами. Однак це не кращий варіант отримання необхідних даних, оскільки поліси містять конфіденційну інформацію, а власне обробка позовів – це складний, тривалий процес, який не відрізняється інформаційною повнотою стосовно поставленої задачі. На сьогодні для математичного опису усіх типів ризиків у страхуванні широко застосовують методи прикладної статистики, теорії ймовірностей та нечіткої логіки.

У процесі аналізу ситуацій, пов'язаних із виникненням фінансового ризику, важливо отримати об'єктивну інформацію стосовно поточного стану страхової компанії з незалежних джерел. Необхідно встановити коректність дій відділу ризик-менеджменту, дослідити інформацію, яку він використовує, та методи її обробки. Однією із найважливіших характеристик цієї інформації є її повнота – чи достатньо наявних даних для побудови моделі з метою оцінювання і прогнозування можливих втрат. Часто може виявитись, що наявної інформації недостатньо для виявлення усіх типів ризиків, які виникають у компанії. Так, відомі втрати можуть не відображати усіх типів ризиків, з якими прийшлось зіткнутись компанії, оскільки деякі ризики ще не призвели до втрат, а тому були проігноровані.

У таких випадках необхідно виконати додатковий аналіз, який стосується: 1 - уточнення моменту часу, коли виникли фінансові втрати, а також встановлення факту прийняття рішення, яке призвело до цих втрат; 2 - оцінювання можливого доходу, який могла отримати компанія у випадку уникнення ризикової ситуації; 3 - розподілу наявних фінансових втрат відповідно до впливу декількох факторів ризику, якщо такі фактори існують (тобто кількість існуючих факторів ризику більша одиниці); 4 - збору додаткової інформації від персоналу,

який має відношення до ризикової ситуації, що мала місце. Метою додаткового аналізу є вияснення послідовності подій, які фактично призвели до втрат, уточнення причин виникнення цих подій та встановлення можливостей їх уникнення; а також встановити – чому не вдалось їх уникнути. Завжди існує ймовірність того, що не всі втрати були виявлені і враховані відділом ризик-менеджменту. Тому робота з персоналом може надати додаткову інформацію стосовно інших можливих втрат, які вдалось відвернути або ж вони залишаються цілком актуальними.

Загалом процедура ідентифікації типів ризиків та управління ними може бути представлена у вигляді циклічної послідовності таких дій: 1 – встановлення можливих типів ризиків для компанії; 2 – ідентифікація, поглиблене розуміння та опис ситуацій, які сприяють виникненню факторів ризиків; 3 – докладний аналіз типів можливих ризиків із встановленням мір втрат та методів їх оцінювання і прогнозування; 4 – прийняття управлінських рішень стосовно контролю (менеджменту) ризиків конкретних типів; 5 – спостереження за виконанням управлінських рішень, виявлення і аналіз індикаторів настання можливих ризиків; 6 – складання докладного звіту стосовно виконаних дій, спрямованих на уникнення, ігнорування або активне управління ситуаціями з виникненням ризиків. Реалізація вказаної циклічної процедури ідентифікації ризиків та менеджменту відповідних ситуацій повинна спиратись на класифікацію можливих ризиків для компанії (підприємства). Часто вживають термін „*ризик підприємства*", який включає всі можливі ризики для конкретного підприємства. Ризик підприємства, у свою чергу, розділяють на *основний ризик для бізнесу* і *операційний ризик*, які також мають свої складові.

Існує декілька методів кількісного аналізу, які можна застосувати до розв'язання задачі поглибленого розуміння суті та оцінювання рівня фінансового ризику. До методів цього класу відносять такі: 1 – *статистичне оцінювання* (емпіричні дослідження, оцінювання максимально можливих втрат, оцінювання функцій розподілу ймовірностей; регресійний аналіз); 2 – *частотний аналіз втрат* (частотний аналіз величини втрат, теорія екстремальних значень, стохастичні диференціальні рівняння); 3 – *статистичний байєсівський підхід* (моделі динаміки досліджуваних об'єктів, діаграми впливу, байєсівські мережі довіри і моделі причинних зв'язків, аналіз карт розвитку процесів); 4 – *системи штучного інтелекту* на основі нейронних мереж і нейронечітких моделей (класифікація клієнтів та підприємств, оцінювання рівня можливих втрат); 5 – *моделі на основі методів Монте - Карло* і *моделі з перемиканням режимів* (генерування сценаріїв розвитку, аналіз доходності, стратегічний інвестиційний аналіз); 6 – *експертне оцінювання і нечітка логіка* (методи нечіткої логіки, безпосереднє оцінювання правдоподібності варіантів, метод Делфі, моделі активів капіталу та ціноутворення, оцінювання ринкового ризику у страхуванні); 7 – *практичні методики менеджменту ризиків* (стрес тестування і аналіз сценаріїв, промислово-бізнесові сценарії, динамічний фінансовий аналіз, ринкове бета-порівняння окремих компаній між собою у межах секторів ринку).

Математичні моделі, які використовуються в актуарній математиці, класифікують як детерміністичні та стохастичні. Очевидно, що моделі – це спрощене представлення можливих наслідків майбутніх невизначених подій. Невизначеність пов'язана у даному випадку з часом виникнення та можливими наслідками. Хоча при використанні детерміністичних моделей оцінка прогнозу генерується цілком визначено, ця „визначеність" ґрунтується на припущеннях, які невизначені за своєю природою. Це стосується типів розподілів відповідних випадкових величин, вибору методів оцінювання структури і параметрів моделі, оцінювання типу випадкового збурення і т. ін. Якщо припущення, прийняті при побудові детерміністичної моделі, відповідають фактичній поведінці досліджуваного процесу і майбутні зміни цього процесу відповідають оцінкам прогнозів за моделлю, то такими оцінками можна користуватись при прийнятті рішень. Стохастичні моделі спрямовуються на прогнозування ймовірностей виникнення подій також на основі інформації стосовно поведінки процесів у минулому. Вони ґрунтуються на гіпотезах і припущеннях, які є логічно узгодженими стосовно ймовірного протікання подій у майбутньому з урахуванням можливих невизначеностей. Необхідно зазначити, що прогнози, отримувані за такими моделями, не можуть бути повністю визначеними, оскільки повна визначеність фактично означає непридатність моделі для практичного використання.

Байєсівський підхід до опису ризиків. Нехай $\mathbf{M}=\{M_1, M_2, ..., M_n\}$ – множина моделей, які застосовуються до опису ризиків; $p(x \mid M_i, \theta_i)$ – функція правдоподібності для моделі M_i, $i=1,...,N$ з параметрами θ_i і наявними даними x; і нехай $p(\theta_i \mid M_i)$ – апріорний розподіл вектора параметрів θ_i вибраної моделі M_i. Апостеріорний розподіл параметрів θ_i за умови відомої структури моделі M_i і даних x можна записати так: $p(\theta_i \mid M_i, x) = c_i\, p(x \mid M_i, \theta_i)\, p(\theta_i \mid M_i)$, де c_i – нормуюча константа. Структури різних моделей відрізняються кількістю параметрів, при цьому перевага, за принципом економії, надається простішим моделям, якщо якість математичного опису залишається прийнятною. Для вибору кращої моделі можна скористатись інформаційним критерієм Акайке (ІКА), модифікованим для даного класу імовірнісних моделей [9]:

$$\log \tilde{p}(x \mid M_i, \theta_i) = \log p(x \mid M_i, \theta_i) - A(k_i),$$

де k_i – розмірність вектора θ_i; $A(k_i) = k_i$ – зростаюча функція k_i. Він приймає максимальне значення для кращої моделі. Цей критерій можна представити у формі, яка мінімізується для кращої моделі, тобто $-2 \log p(x \mid M_i, \theta_i) + 2k_i$. Альтернативою для ІКА є критерій Байєса-Шварца, у якому $A(k_i) = 0{,}5 k_i \log N$, де N – потужність вибірки даних x. Дещо інші модифікації цих критеріїв використовують також для аналізу якості регресійних моделей різноманітних структур.

Після вибору структури моделі $M^* = M_{i^*}$ необхідно максимізувати апостеріорну щільність $p(\theta_{i^*} \mid M_{i^*}, x)$ по θ_{i^*} (або обчислити апостеріорне середнє $E[\theta_{i^*} \mid M_{i^*}, x]$) для того, щоб знайти кращу оцінку вектора параметрів $\hat{\theta}_{i^*}$. Якщо потужність вибірки даних достатньо велика, а апріорний розподіл $p(\theta_{i^*} \mid M_{i^*})$ – дифузійний, то апостеріорний максимум можна замінити оцінкою максимальної правдоподібності. Знайдені структура M_{i^*} і параметри моделі θ_{i^*} надалі вважаються прийнятними. Розглянемо тепер можливість введення невизначеностей у параметри і структуру моделі.

Отримаємо вирази для апостеріорних розподілів для M_i і $\theta_i \mid M_i$ (для спрощення записів нижній індекс "i" не будемо застосовувати). Позначимо через $p_r(M)$ апріорну ймовірність оцінювання структури моделі M, а через $p(\theta_i \mid M_i)$ – апріорний розподіл параметрів θ за умови відомої структури M. Згідно з правилом Байєса маємо:

$$p(\theta, M \mid x) = p(\theta \mid m, x)\, p_r(M \mid x) = p(\theta \mid m, x) \cdot c \cdot p_r(M)\, p(x \mid M),$$

тут $p(x \mid M) = \int p(x \mid \theta, M)\, p(\theta \mid M)\, d\theta$; c – нормуюча константа. Таким чином, спільна апостеріорна щільність для (θ, M) визначається добутком апостеріорної щільності для θ за умови, що модель M коректна, і апостеріорної ймовірності визначення коректної структури моделі M при наявності даних x. Спільний розподіл (θ, M) визначається за виразом:

$$p(\theta, M \mid x) = c\left(c_i\, p(x \mid \theta, M)\, p(\theta \mid M)\right) p(x \mid M)\, p_r(M),$$

де c_i – нормуюча константа для $p(\theta \mid M, x)$, а c – нормуюча константа для всього апостеріорного розподілу.

Припустимо, що необхідно знайти ймовірність банкротства y, яка визначається через параметри моделі M. Умовна ймовірність для y має вигляд:

$$p(y \mid x) = \sum_i p(y \mid x, M_i)\, p_r(M_i \mid x),$$

де

$$p(y \mid x, M_i) = \int p(y \mid x, M_i, \theta_i)\, p(\theta_i \mid M_i, x)\, d\theta_i;$$

$$p(\theta_i \mid M_i, x) = c_i\, p(x \mid M_i, \theta_i)\, p(\theta_i \mid M_i).$$

Асимптотичний аналіз розподілу $p(\theta_i \mid M_i, x)$ свідчить [10, 11], що

$$\log p(\theta_i \mid M_i, x) = c - \frac{1}{2}(\theta_i - \hat{\theta}_i)^T \mathbf{H}_i (\theta_i - \hat{\theta}_i) + o(|\theta_i - \hat{\theta}_i|^2),$$

де $\hat{\theta}_i$ – зважене середнє оцінки максимальної правдоподібності для θ_i і моди апріорного розподілу для θ_i; $\mathbf{H}_i$ – сума гессіанів у відповідному максимумі функцій правдоподібності та апріорної щільності.

Якщо апріорний розподіл дифузійний і потужність вибірки N достатньо велика, то при обчисленні оцінок можна застосувати деяку прийнятну апроксимацію, тобто матрицю $\mathbf{H}_i$ можна апроксимувати добутком $N\hat{\mathbf{B}}_i$, де $\hat{\mathbf{B}}_i = \mathbf{B}_i(\hat{\theta}_i)$ – інформаційна матриця для одного спостереження за умови, що $\hat{\theta}_i$ – це істинне значення для θ_i. Оцінку $\hat{\theta}_i$ можна обчислити також за методом максимальної правдоподібності.

Розглянемо тепер апостеріорний розподіл для моделі M_i:

$$p_r(M_i \mid x) = c\, p_r(M_i)\, p(x \mid M_i),$$

де $p(x \mid M_1), p(x \mid M_2), \ldots$ – фактори Байєса (з точністю до масштабної константи). Дещо спрощений розрахунок цих факторів можна виконати за формулою [12]:

$$\log p(x \mid M_i) = c + \frac{1}{2} k_i \log 2\pi - \frac{1}{2}\log\left|\hat{\mathbf{I}}_i\right| + \log p(x \mid \hat{\theta}_i, M_i) + \log p(\hat{\theta}_i \mid M_i) + O(N^{-1}). \quad (1)$$

Тут $\hat{\mathbf{I}}_i$ – інформаційна матриця для даних x, які описуються моделлю з параметрами $\hat{\theta}_i$; k_i – розмірність вектора θ_i; $p(\theta_i \mid M_i)$ – апріорний розподіл для θ_i. Оскільки спостереження за припущенням незалежні, то $\hat{\mathbf{I}}_i = N\hat{\mathbf{B}}_i$, де $\hat{\mathbf{B}}_i$ – інформаційна матриця, що відповідає одному спостереженню при використанні моделі M_i з параметрами $\hat{\theta}_i$. Таким чином, можна записати, що $\log\left|\hat{\mathbf{I}}_i\right| = k_i \log N + \log\left|\hat{\mathbf{B}}_i\right|$.

Зазначимо, що при зростанні N другий член справа буде залишатись приблизно постійним для кожної моделі. Якщо k_i прийматиме одне і те ж значення для всіх моделей, то змінюватись буде тільки $\log\left|\hat{\mathbf{B}}_i\right|$. Вплив на критерій складової $\log p(\hat{\theta}_i \mid M_i)$, пов'язаної з апріорним розподілом, незначний, особливо при використанні дифузійного розподілу. Після введення позначення $\hat{l}_i = \log p(x \mid \hat{\theta}_i, M_i)$ і вилучення члена $\log p(\hat{\theta}_i \mid M_i)$ критерій (1) прийме спрощений вигляд:

$$\log p(x \mid M_i) \approx c + \frac{1}{2} k_i \log 2\pi + \hat{l}_i - \frac{1}{2} k_i \log N - \frac{1}{2}\log\left|\hat{\mathbf{B}}_i\right|, \quad (2)$$

або

$$\log p_r(M_i \mid x) \approx \log p_r(M_i) + \frac{1}{2} k_i \log 2\pi + \hat{l}_i - \frac{1}{2} k_i \log N - \frac{1}{2}\log\left|\hat{\mathbf{B}}_i\right| + c, \quad (3)$$

де c – нормуюча константа, яка забезпечує рівність: $\sum_i p_r(M_i \mid x) = 1$.

Приклад побудови моделі випадкових надходжень (платежів). Нехай $\{x(k)\}$ – випадковий процес надходження платежів, де k – дискретний час надходжень. Отже, накопичення на перший момент часу складають $\exp(x(1))$, а на довільний момент k накопичення складуть $\exp(x(1) + x(2) + \ldots + x(k))$. Для зручності аргумент під експонентою позначимо так: $y(k) = \sum_{i=1}^{k} x(i)$. Необхідно визначити типи розподілів для

$y(k)$ і $F(k) = \exp(y(k))$. Однією із простих моделей, які використовують для опису подібних фінансових процесів, є авторегресія першого порядку, тобто рівняння АР(1):

$$x(k) = a_0 + a_1 x(k-1) + \varepsilon(k), \tag{4}$$

де $\varepsilon(k)$ – процес випадкових збурень, який приймемо у даному випадку нормальним без особливого порушення загальності аналізу. Рівняння (4) представимо у зручнішій для подальшого аналізу формі:

$$x(k) - \mu = a(x(k-1) - \mu) + \sigma z(k), \tag{5}$$

тут μ – середнє значення відповідного ряду даних; a, σ - параметри моделі; $\{z(k)\} \sim \text{Norm}(0, 1)$ – послідовність незалежних випадкових величин, які мають стандартний нормальний розподіл. Зазначимо, що рівняння (5) – це наближений дискретний аналог звичайного диференціального рівняння першого порядку. Знайдемо вирази для оцінок параметрів за методом максимальної правдоподібності.

Запишемо функцію умовної правдоподібності для ряду надходжень платежів:

$$f(x \mid \mu, \sigma^2, a) =$$

$$= \prod_{i=-N+2}^{0} \left\{ (2\pi\sigma^2)^{-1/2} \exp\left[-\frac{1}{2\sigma^2} \left(x(i) - \mu - a(x(i-1) - \mu) \right)^2 \right] \right\}. \tag{6}$$

За допомогою функції (6) знайдемо вирази для оцінок параметрів моделі (5):

$$\hat{a} = \frac{\sum_{i=-N+2}^{0} \left(x(i) - \hat{\mu} \right) \left(x(i-1) - \hat{\mu} \right)}{\sum_{i=-N+2}^{0} \left(x(i-1) - \hat{\mu} \right)^2};$$

$$\hat{\mu} = \frac{1}{N-1} \left(\sum_{i=-N+1}^{-1} x(i) + \frac{x(0) - x(-N+1)}{1 - \hat{a}} \right);$$

$$\sigma^2 = \frac{1}{N-1} \sum_{i=-N+2}^{0} \left[x(i) - \hat{\mu} - \hat{a}(x(i-1) - \hat{\mu}) \right]^2.$$

Запишемо наближений вираз для функції умовної правдоподібності (6) розкладанням в ряд:

$$f(x \mid \mu, \sigma^2, a) \propto$$

$$\propto (\sigma^2)^{-(N-1)/2} \exp\left[-\frac{1}{2\sigma^2} \left\{ \phi_1 + \phi_2 (\mu - \hat{\mu})^2 + \phi_3 (a - \hat{a})^2 + \right.\right.$$

$$+ \phi_4 (\mu - \hat{\mu})(a - \hat{a}) + \phi_5 (\mu - \hat{\mu})(a - \hat{a})^2 +$$

$$\left.\left. + \phi_6 (\mu - \hat{\mu})^2 (a - \hat{a}) + \phi_7 (\mu - \hat{\mu})^2 (a - \hat{a})^2 \right\} \right],$$

де "$\propto$" – знак пропорційності;

$$\phi_1 = (N-1)\hat{\sigma}^2;\ \phi_2 = (N-1)(1-\hat{a})^2;\ \phi_3 \approx (N-1)\hat{\sigma}^2/(1-\hat{a}^2);$$

$$\phi_4 = 2(x(-N+1) - x(0)) \approx 0;\ \phi_5 = \phi_4/(1-\hat{a}) \approx 0;\ \phi_6 = -2(N-1)(1-\hat{a});$$

$$\phi_7 = (N-1).$$

Для випадку розв'язання задачі моделювання ризику у наведеній вище формі запропоновано такий апріорний розподіл [13, 14]:

$$f(\mu, \sigma^2, a) = \sigma^{-3} (1-a)^{1/2} (1+a)^{-1/2},$$

або $f(\mu,\sigma^2,a)=\sigma^{-3}(1-a)^{3/2}(1+a)^{1/2}$, які можна застосовувати при $a\neq\pm1$. Тепер можна записати такі апостеріорні розподіли для параметрів процесу:

$$(\mu|x,\sigma^2,a)\sim \mathrm{Norm}\left(\hat{\mu},\frac{\sigma^2/(N-1)}{(1-a)^2}\right);$$

$$f(a|x)=d(\hat{a},N)\left[1-\hat{a}^2+(a-\hat{a})^2\right]^{-(N-1)/2}(1-a^2)^{-1/2},\quad -1<a<1.$$

Враховуючи знайдені параметри μ, σ^2, a за умови наявності даних $x(k)$, повернемось до рівняння (5):

$$x(i)-\mu=a(x(i-1)-\mu)+\sigma z(i)=$$
$$=\sigma(z(i)+az(i-1)+...+a^{s-1}z(1))+a^s(x(0)-\mu).$$

Наведені результати дають можливість отримати вираз для $y(k)$:

$$y(k)|\mu,\sigma^2,a,x=\mu k+(x(0)-\mu)\,M(a,k)+\sigma(V(a,k))^{1/2}Z,$$

де $M(a,k)=a(1-a^k)/(1-a)$;

$$V(a,k)=\frac{1}{(1-a)^2}\left(k-\frac{2a(1-a^k)}{1-a}+\frac{a^2(1-a^{2k})}{1-a^2}\right).$$

Для виконання обчислювального експерименту вибрано дані стосовно надходження платежів до вибраної страхової компанії за п'ятирічний період. В результаті отримано такі оцінки параметрів:

$$\hat{a}=-0{,}2587;\ \ \hat{\mu}=0{,}2165;\ \ \hat{\sigma}^2=0{,}0873,$$

а оцінки, отримані за методом моментів для цих статистичних даних, мають такі значення: $\hat{\mu}=0{,}2139;\ \ \hat{\sigma}^2=0{,}0861$. Апріорна ймовірність для моделі вибрана рівною 0,5, що логічно в умовах відсутності додаткової інформації стосовно цієї оцінки. Апостеріорна ймовірність для моделі склала: $p_г(M)=0{,}59$. Таким чином, отримано параметри прогнозуючого апостеріорного розподілу в умовах наявності параметричної і статистичної невизначеностей досліджуваного процесу (у цьому полягає перевага ймовірнісного підходу). Очевидно також, що отримана апостеріорна ймовірність для моделі в цілому недостатньо висока, що можна пояснити використанням спрощеної моделі авторегресії першого порядку та наближеним обчисленням функції умовної правдоподібності. Також потребує поглибленого дослідження вплив типу апріорного розподілу на остаточний результат. Поглиблений аналіз можливості застосування імовірнісних моделей розглянутого типу потребує застосування декількох альтернативних підходів до побудови моделі і порівняння результатів застосування кожної з них.

3. Висновки

Існує широка множина актуарних ризиків, які потребують аналітичного дослідження за допомогою математичних і статистичних моделей різної структури і складності. Визначено методи кількісного аналізу, які можна застосувати до розв'язання задачі поглибленого розуміння суті та оцінювання рівня фінансового ризику. Для оцінювання рівня та ймовірності втрат можна успішно застосовувати моделі імовірнісного типу, оскільки вони дають можливість враховувати параметричні і статистичні невизначеності досліджуваного процесу. Розглянуто байєсівський підхід до опису ризиків і запропоновано процедуру формування моделі імовірнісного типу, яка використана для побудови прогнозуючої моделі стосовно надходжень платежів. Виконано обчислювальні експерименти з метою оцінювання параметрів прогнозуючого розподілу. Отримані результати близькі до результатів застосування методу моментів до фактичних статистичних даних. Встановлено, що апостеріорна ймовірність для моделі склала: $p_г(M)=0{,}59$, тобто отримана апостеріорна ймовірність

для моделі в цілому недостатньо висока, що можна пояснити використанням спрощеної моделі авторегресії першого порядку та наближеним обчисленням функції умовної правдоподібності.

У подальших дослідженнях необхідно застосувати множину альтернативних моделей-кандидатів імовірнісного типу з метою вибору кращої для отримання високоякісних оцінок прогнозів. Для побудови високоефективних прогнозуючих моделей перспективним є застосування методу Монте-Карло для марковських ланцюгів. У даному випадку марковськими ланцюгами представляють невідомі оцінки параметрів моделей. Такий чисельний підхід надає можливість суттєво збільшити кількість параметрів і застосувати ускладнені математичні моделі.

Список літератури: 1. *Bernstein P.L.* Against the Gods: the remarkable story of risk / P.L. Bernstein . New York: John Wiley & Sons, Inc., 1996. 383 p. **2.** *Згуровский М.З*. Системный анализ / М.З. Згуровский, Н.Д. Панкратова. Киев: Наук. думка, 2011. 900 с. **3.** *Holsapple C.W.* Decision support systems / C.W.Holsapple, A.B.Winston. Saint Paul (USA): West Publishing Company, 1996. 850 p. **4.** *Aven T.* On risk defined as an event where the outcome is uncertain / T.Aven, O. Renn // Journal of Risk Research. 2009. No. 12. P. 1 – 11. **5.** *Kaplan S.* The words of risk analysis / S.Kaplan // Risk Analysis, 1997. Vol. 17. P. 407 – 417. **6.** *Actuarial* Mathematics / [Bowers N.L., Gerber H.U., Hickman J.C., Jones D.A., Nesbitt C.J.]. Itasca (Illinois): The Society of Actuaries. 1986. 624 p. **7.** *Шахов В.В.*Теория и управление рисками в страховании / В.В.Шахов, В.Г. Медведев, А.С. Миллерман. М.: Финансы и статистика, 2002. 224 с. **8**. *Фалин Г.И.*Актуарная математика в задачах / Г.И. Фалин, А.И.Фалин. М.: Физматлит, 2003. 192 с. **9.** *Sik-Yum Lee.* Structural Equation Modeling. – New York: Wiley & Sons, Ltd, 2007. 460 p. **10.** *Bernardo J.M.* Bayesian theory / J.M.Bernardo, A.F. Smith. – New York: John Wiley & Sons, Inc., 2001. 586 p. **11.** *Bayesian* data analysis / [Gelman A., Carlin J.B., Stern H.S., Rubin D.B.]. New York: Chapman and Hall/CRC, 2004. 670 p. **12.** *Draper D.* Assessment and propagation of model uncertainty / D.Draper // Journal of the Royal Statistical Society.1995. Ser. B. Vol. 57. P. 45 – 97. **13.** *Klugman S.* Bayesian statistics in actuarial science / S. Klugman. Boston: Kluwer, 1992. 256 p. **14**. *Rossi P.E. R.* Bayesian statistics and marketing / P.E. Rossi, G.M. Allenby, R. McCulloch. New Jersey: John Wiley & Sons, Ltd, 2005. 348 p.

Надійшла до редколегії 11.06.2012

Кожухівська Ольга Андріївна, канд. техн. наук, старший викладач кафедри інформатики та інформаційної безпеки Черкаського державного технологічного університету. Наукові інтереси: аналіз і моделювання складних систем. Адреса: Україна, 18006, Черкаси, бульвар Шевченка, 460, тел. 0472 730217. E-mail: olga-kozhuhovska@mail.ru.

УДК 681.324:519.613

К.Е. ГЕРАСИМЕНКО

МЕТОД ПОВЫШЕНИЯ КОНТРОЛЕПРИГОДНОСТИ КРИТИЧЕСКИХ СИСТЕМ УПРАВЛЕНИЯ АЭС

Разрабатывается и тестируется на прикладных примерах метод контролепригодности оборудования защит из состава управляющей системы безопасности АЭС. Отличительной особенностью метода от существующих [1-16] является использование логических элементов защит, построенных на базе арифметических операций с интегральной оценкой значений входных сигналов в диапазоне [0;1], без применения логических операций и операций отношения. Это позволяет контролировать работоспособность данных элементов по их реакции на изменения входного непрерывного сигнала от канала ввода в АЦП через все логические элементы защит, в которых используется данный сигнал, до дискретного выходного элемента, формирующего команду защиты на конкретный исполнительный механизм. Данный метод позволяет обеспечить контроль и диагностирование целого ряда неисправностей типа «несрабатывание», относящихся к категории скрытых в существующих реализациях оборудования защит, которые применяют логические операции и операции отношения.

1. Введение

Одним из основных показателей, характеризующих надежность оборудования защит из состава управляющих систем безопасности (УСБ) атомных электростанций (АЭС), является вероятность правильного выполнения дискретной функции по формированию последо-

вательности команд защитных действий с учетом наличия отказов типа “несрабатывание”. Критерием такого вида отказа является отсутствие команды защиты при наличии «исходного» события, т.е. при появлении на входах оборудования защит УСБ любой совокупности данных, которая должна вызвать формирование команды.

2. Постановка задачи исследования

Ввиду того, что отказ типа «несрабатывание» для УСБ в целом может быть причиной возникновения нештатной ситуации или аварии, разработка методов контроля и диагностирования, позволяющих выявлять такого рода отказы, является актуальной задачей и предметом различного рода исследований и конструкторских решений. В общем случае к отказам УСБ данного типа могут приводить комбинации как однотипных (отказы по общей причине), так и разнотипных видов скрытых неисправностей в резервированных компонентах УСБ, имеющих временную корреляцию [1-9].

Данная проблема актуальна для всех типов оборудования, реализующего дискретные функции, независимо от используемой элементной базы и принципов построения: аналоговые приборы на транзисторах или реле, программно-логические интегральные схемы (ПЛИС), микропроцессоры с инструкциями в виде программного кода.

Цель исследования – разработка математического аппарата, позволяющего использовать естественные «фоновые» флуктуации входных аналоговых сигналов в процессе штатной работы АЭС для непрерывного автоматического контроля всех элементов УСБ: от АЦП до формирования команд на исполнительные механизмы через логические блоки алгоритма защит.

3. Математический аппарат метода

Существующие методы решения данной проблемы описаны в [10-14] и сводятся к проверке работоспособности элементов оборудования защит путем контроля их реакции на специальные тестовые воздействия, поскольку использование данных о рабочих воздействиях от объекта, как правило, недостаточно. Эти методы реализуют контроль работоспособности на срабатывание либо отдельных блоков и устройств, участвующих в реализации функции защит, либо всего оборудования защит или его части (как минимум инициирующей части защит). Методы обоих типов имеют ряд существенных ограничений и недостатков, подробно рассмотренных и проанализированных в [15].

При этом основное ограничение вытекает из самой структуры элементов оборудования защит («>», «<», «и», «или», «2 из 4-х»), построенных на базе дискретных функций, с выходом, определяемым только двумя состояниями 0 (режим ожидания) или 1 (срабатывание защиты). Это в принципе не позволяет обеспечить выполнение непрерывного контроля работоспособности данных элементов на срабатывание. Метод, предполагающий изменение структуры элементов защит, как средство повышения эффективности контроля (проверки) и диагностирования скрытых неисправностей на несрабатывание, предложен в [15,16]. Основная идея данного метода – функциональный элемент защит на базе арифметических операций формирует значение на всем диапазоне от 0 до 1. При этом в нем отсутствуют какие-либо ветвления (условные переходы), определяющие отличия режима ожидания от режима срабатывания защит. Функция работает одинаково в обоих из этих режимов, что позволяет непрерывно контролировать работоспособность соответствующего функционального элемента. Все функции, которые используются в элементах, строятся на базе арифметических операций (сложение, вычитание, умножение, деление), без использования логических операций и операций отношения, за исключением выходных пороговых элементов формирования команд на исполнительные механизмы, что связано с физическими принципами работы их приводов.

В настоящей работе представлены результаты дальнейшего исследования метода, предложенного в [15-16], в части разработки математического аппарата на базе работ [17-19], позволяющего использовать естественные «фоновые» флуктуации входных аналоговых сигналов в процессе штатной работы АЭС для функционального диагностирования логических элементов, реализующих алгоритм защит.

Базис данного аппарата приведен в таблице и состоит из следующих основных подходов:

Типовые логические операции и эквивалентные арифметические операции

Логические операции $x_1,....,x_n, y \in Z_{0-1} = \{0;1\}$ т.е. вход и выход – целые числа 0 или 1	Эквивалентные арифметические операции $x_1,....,x_n, y \in R_{0-1} = \{0,....,1\}$ т.е. x, y – вещественные числа в диапазоне [0;1] 1) сравнение «>=» $x = \frac{I - R_{Imin}}{P_{\geq} - R_{Imin}}$ 2) сравнение «<=» $x = \frac{R_{Imax} - I}{R_{Imax} - P_{\leq}}$ I – значение измеряемого параметра; $P_{\leq}, P_{\geq}$ - пороговое значение (уставка) R_{Imin}, R_{Imax} - нижний и верхний пределы измерения;	
	Нелинейный вид	Интегральный вид
AND $y = (x_1 \wedge x_2 \wedge ..... \wedge x_n)$	$y = x_1 * x_2 * ..... * x_n$	$y = \frac{x_1 + ... + x_n}{n}$
Non-AND $\overline{y = (x_1 \wedge x_2 \wedge ..... \wedge x_n)}$ в форме без инверсии по выходу: $y = (x_1^{/} \vee x_2^{/} \vee ..... \vee x_n^{/})$ $x_1^{/} = 1 - x_1; ....; x_n^{/} = 1 - x_n$ изменяется знак уставки по входным параметрам	$y = 1 - x_1 * x_2 * ..... * x_n$ $y = 1 - (1 - x_1^{/}) * (1 - x_2^{/}) * ....$ $.... * (1 - x_n^{/})$	$y = \frac{1 - x_1 * .... * x_n}{1 - x_1 * .... * x_n + D0}$ $D0 \to 0$ защита от деления на 0 (пример : 10^{-38} - минимально значение float 4 байта) $y = 1 - \left(1 - \frac{x_1^{/} + ... + x_n^{/}}{n}\right) * \frac{(1 - x_1^{/}) * .... * (1 - x_n^{/})}{(1 - x_1) * .... * (1 - x_n) + D0}$
OR $y = (x_1 \vee x_2 \vee ..... \vee x_n)$	$y = 1 - (1 - x_1) * (1 - x_2) * ....$ $..... * (1 - x_n)$	$y = 1 - \left(1 - \frac{x_1 + ... + x_n}{n}\right) * \frac{(1 - x_1) * .... * (1 - x_n)}{(1 - x_1) * .... * (1 - x_n) + D0}$
Non-OR $\overline{y = (x_1 \vee x_2 \vee ..... \vee x_n)}$ в форме без инверсии по выходу: $y = (x_1^{/} \wedge x_2^{/} \wedge ..... \wedge x_n^{/})$ $x_1^{/} = 1 - x_1; ....; x_n^{/} = 1 - x_n$ изменяется знак уставки по входным параметрам	$y = (1 - x_1) * (1 - x_2) * .....$ $..... * (1 - x_n)$ $y = x_1^{/} * x_2^{/} * ..... * x_n^{/}$	$y = \frac{(1 - x_1) * .... * (1 - x_n)}{(1 - x_1) * .... * (1 - x_n) + D0}$ $y = \frac{x_1^{/} + ... + x_n^{/}}{n}$
XOR $y = (x_1 \vee .... \vee x_n) \wedge$ $\overline{\wedge (x_1 \wedge .... \wedge x_n)}$	$y = (1 - (1 - x_1) * (1 - x_2) * .... * (1 - x_n)) *$ $* (1 - x_1 * x_2 * .... * x_n)$	$y = \left(1 - \left(1 - \frac{x_1 + ... + x_n}{n}\right) * \frac{(1 - x_1) * .... * (1 - x_n)}{(1 - x_1) * .... * (1 - x_n) + D0}\right) *$ $* \frac{1 - x_1 * .... * x_n}{1 - x_1 * .... * x_n + D0}$
«2oo3» (2 из 3-х) $y = (x_1 \wedge x_2) \vee$ $\vee (x_1 \wedge x_3) \vee$ $\vee (x_2 \wedge x_3)$	$y = 1 - (1 - x_1 * x_2) *$ $* (1 - x_1 * x_3) *$ $* (1 - x_2 * x_3)$	$y = 1 - \left(1 - \frac{x_1 + x_2 + x_3}{3}\right) *$ $* \frac{(1 - x_1 * x_2) * (1 - x_1 * x_3) * (1 - x_2 * x_3)}{(1 - x_1 * x_2) * (1 - x_1 * x_3) * (1 - x_2 * x_3) + D0}$

1) логические операции заменяются арифметическими с диапазоном входных и выходных переменных в вещественном формате от 0 до 1; входные аналоговые сигналы также нормируются к диапазону от 0 до 1, где 0 соответствует нижней (для уставки «>=») или верхней (для уставки «<=») предельной границе(диапазону) измерения параметра, а 1 соответствует значению уставки (пороговое значение измеряемого параметра); таким образом, значения в диапазоне [0;1) – соответствуют нормальному режиму работы (режим ожидания), а значение 1 – соответствует нарушениям (режим аварии);

2) нелинейный вид арифметического представления стандартных логических операций не является приемлемым для систем критического применения, поскольку может давать потерю значащих разрядов при множественных операциях умножения значений, близких к нулю; для устранения данного эффекта используется интегральное значение (среднее арифметическое) от входных переменных;

3) логические операции, содержащие инверсию выхода, ограничивают возможности диагностирования соответствующих элементов алгоритма в режиме ожидания, поскольку инверсия от любых значений в диапазоне [0;1) всегда дает 1; в связи с этим операции инверсии в алгоритмах защит заменяются на эквивалентные логические операции без инверсии за счет изменения знака уставки (порогового) значения входных сигналов.

4. Заключение

Научная новизна и практическая значимость. Разработанный метод повышения контролепригодности оборудования защит, использующий функциональные элементы на базе арифметических операций, характеризуется такими особенностями:

1) обеспечивает диагностирование следующих видов скрытых неисправностей типа «несрабатывание»: дефекты функциональных элементов, характеризуемые несоответствием значений входных и значений выходных переменных проектному алгоритму; неисправности связей между функциональными элементами, характеризуемые отсутствием или искажением данных между источником и приемником;

2) повышает контролепригодность путем замены логических операций на арифметические с диапазоном входных и выходных переменных в вещественном формате от 0 до 1, при этом значения в диапазоне [0;1) – соответствуют нормальному режиму работы (режим ожидания), а значение 1 – соответствует нарушениям (режим аварии);

3) обеспечивает контроль прохождения любого изменения значения входного сигнала в пределах разрешающей способности используемых АЦП от входа через все функциональные элементы («сравнение с пороговым значением», «и», «или», «2 из 3» и др.), в которых участвует данный сигнал, до дискретного элемента управления исполнительным механизмом.

Список литературы: **1.** *Безопасность* атомных станций. Информационно-управляющие системы / М.А. Ястребенецкий, В.Н. Васильченко, С.В. Виноградская и др. К.: Техника, 2004. 470 с. **2.** *Instrumentation* and control systems important to safety in Nuclear Power Plants: Nuclear Energy Series / International Atomic Energy Agency. Vienna: IAEA, 2002. No. NS-G-1.3. 91 p. **3.** *Safety* of Nuclear Power Plants: Design, Safety Standards Series / International Atomic Energy Agency. Vienna: IAEA, 2000. No. NS-R-1. 125 p. **4.** Software for Computer Based Systems Important to Safety in Nuclear Power Plants: Safety Standards Series / International Atomic Energy Agency. Vienna: IAEA, 2000. No. NSG-1.1. 150 p. **5.** *International* Electrotechnical Commission (IEC) 60880 – 2004, Nuclear Power Plants — Instrumentation and Control Systems Important to Safety — Software Aspects for Computer-Based Systems Performing Category A Functions. **6.** *International* Electrotechnical Commission (IEC) 60987 – 2007, Nuclear Power Plants – Instrumentation and Control Important to Safety – Hardware Design Requirements for Computer-Based Systems. **7.** *Institute* of Electrical and Electronic Engineers (IEEE) 7-4.3.2 , Standard Criteria for Digital Computers in Safety Systems of Nuclear Power Generating Stations. **8.** *Макдональд Д.* Промышленная безопасность, оценивание риска и системы аварийного останова: Пер. с англ./ Д. Макдональд. М.: ИДТ, 2007. 409 с. **9.** *Смит Д.* Безотказность, ремонтопригодность и риск: Пер. с англ. / Д. Смит. М.: ИДТ, 2007. 432 с. 10. НП 306.2.141-2008. Общие положения безопасности атомных станций. К: ГКЯРУ, 2008. 42 с. **11.** *Protecting* against common cause failures in Digital I&C Systems of Nuclear Power Plants: Nuclear Energy Series / International Atomic Energy Agency. Vienna: IAEA, 2009. No. NP-T-1.5. 65 p. **12.** *Ястребенецкий М.А.* Информационные и управляющие системы АЭС Украины: результаты и проблемы / М.А. Ястребенецкий // Проблемы обеспечения безопасности информационных и управляющих систем АЭС // Сб. науч. тр. Одесса: Астропринт, 2010. С .9-19. **13.** *Modern* Instrumentation and Control for Nuclear Power Plants: Technical Reports Series / International Atomic Energy Agency. Vienna: IAEA, 1999.

No. 387. 629 p. **14.** *Application* of the Single Failure Criterion: Safety Series / International Atomic Energy Agency. Vienna: IAEA, 1990. No. 50-P-1. 134 p. **15.** *Герасименко К.Е.* Методы непрерывного контроля и диагностирования оборудования управляющих систем безопасности энергоблоков АЭС по функции защит / К.Е. Герасименко // Радіоелектронні і комп'ютерні системи. 2010. №3 (44). С. 152-156. **16.** *Герасименко К.Е.* Использование непрерывных функций в элементах оборудования защит АЭС для диагностирования неисправностей типа «несрабатывание по требованию» / К.Е. Герасименко // Радіоелектронні і комп'ютерні системи. 2011. №1 (49). С. 29-33. **17.** *Бондаренко М.Ф., Кривуля Г.Ф., Рябцев В.Г., Фрадков С.А., Хаханов В.И.* Проектирование и диагностика компьютерных систем и сетей. К.: НМЦ ВО. 2000. 306 с. **18.** *Хаханов В.И., Литвинова Е.И., Чумаченко С.В., Гузь О.А.* Логический ассоциативный вычислитель // Электронное моделирование. 2011. № 1(33). С. 73-89. **19.** *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.

Поступила в редколлегию 01.06.2012

Герасименко Константин Евгеньевич, заведующий отделом информационно-управляющих систем ЧАО СНПО "Импульс". Научные интересы: техническая диагностика цифровых систем управления объектами с повышенными требованиями к безопасности и надежности. Адрес: Украина, 93405, Северодонецк, пл. Победы, 2, тел. 60194. E-mail: gerasymenko.k.e@yandex.ua.

УДК 004.853

О.М. ПОЧАНСКИЙ

ЭКСПЕРТНАЯ СИСТЕМА СЕМАНТИЧЕСКОГО ПОИСКА РЕЛЕВАНТНЫХ ДАННЫХ И ФОРМИРОВАНИЯ АДАПТИВНЫХ WEB-СТРАНИЦ

Рассматривается подход к решению задачи семантического поиска релевантных данных в сети Интернет и построения адаптивных Web-страниц на основе создания универсального программного объекта. Итоговый результат, соответствующий требованиям и интересам пользователя, формируется экспертной системой в виде персонализированного рейтинга Web-документов, который зависит от значений социального индекса.

1. Введение

Под экспертной системой понимают компьютерную интеллектуальную систему, которая эмулирует способность эксперта к принятию решений [1]. Главными преимуществами экспертных систем являются: повышенная доступность (для обеспечения доступа к экспертным знаниям могут применяться любые подходящие компьютерные средства); постоянство (экспертные знания никуда не исчезают); возможность получения экспертных знаний из многих источников (с помощью экспертных систем могут быть собраны знания многих экспертов и привлечены к работе над задачей, выполняемой одновременно и непрерывно); быстрый отклик (при использовании современного аппаратного и программного обеспечения экспертная система может реагировать быстрее и эффективнее, чем эксперт-человек); возможность использования в качестве интеллектуальной базы данных (экспертные системы могут применяться для доступа к базам данных с помощью интеллектуального способа доступа).

Следовательно, разработка экспертной системы для организации эффективного поиска информации является перспективным вариантом решения таких задач, как синтез динамических адаптивных Web-страниц на основе создания универсального программного объекта, реализующего поиск и интеллектуальный анализ данных сети Интернет по предлагаемым методам и критериям [2].

Выделим наиболее существенные моменты, относящиеся к распределению функций поисковой экспертной системы:

– функции эксперта выполняет программный модуль А, реализующий алгоритм поиска информации по социальному индексу, который тесно связан с текущими тематическими интересами пользователя (социальный критерий);

– функции инженера по знаниям выполняет программный модуль B, реализующий алгоритмы выделения значимой информации и оценки Web-документов по их структурным характеристикам (методы обработки);

– координацию работы программных модулей A и B выполняет специализированный программный модуль C, который отвечает за индексирование и обработку Web-ресурсов, найденных в сети Интернет;

– полученные в результате поиска релевантные отфильтрованные данные (явно выраженные знания) поступают в базу знаний экспертной системы, представленную онтологией с заранее определенной структурой.

В данной работе предлагается вариант экспертной поисковой системы, построенной по принципу организации программ, основанных на знаниях, который рассмотрен в [3] с учетом отмеченной выше специфики распределения функций.

2. Формирование базы знаний экспертной поисковой системы

В общем случае под базой знаний понимают особого рода базу данных, предназначенную для оперирования знаниями (мета-данными) [3]. Полноценные базы знаний содержат в себе не только фактическую информацию, но и правила вывода, допускающие автоматическую обработку информации. Как правило, они предназначены для поиска способов решения определенной проблемы из некой предметной области, основанных на записях базы знаний и на пользовательском описании ситуации. Таким образом, любая экспертная система или программа-агент так или иначе взаимодействует с определенной базой знаний, описывающей предметную область.

Рассмотрим процесс организации и текущей модификации базы знаний для системы эффективного поиска Интернет-ресурсов в рамках заданной тематики.

Построение любой базы знаний невозможно без формирования постоянных источников поступления информации. В разрабатываемой экспертной системе поиска в качестве основного источника поступления новых Web-страниц по заданной тематике была предложена специально разработанная настройка над браузером пользователя, называемая плагином. Кроме того, на начальном этапе в качестве дополнительного источника информации предлагается использовать тематические каталоги поисковых систем Яндекс, Mail.ru и Yahoo, что позволяет сформировать основу информационной подпитки разрабатываемой экспертной системы поиска непосредственно уже на первом шаге введения ее в эксплуатацию. При этом исключаются возможные проблемы с корректным выводом релевантных результатов по запросу пользователя из-за отсутствия Web-страниц по запрашиваемой им тематике.

Кроме того, база знаний системы должна накапливать данные, поступающие из внешних источников (Web-страниц) таким образом, чтобы обеспечить вывод необходимой пользователю информации в соответствии с составленным им запросом. При этом главным показателем эффективности работы выполняемого поиска является вывод наиболее близкого релевантного результата в виде списка, состоящего из нескольких Web-страниц, соответствующих тематике сформированного запроса. Этому могут способствовать грамотно спроектированные специализированные правила логического вывода в рамках заданной модели базы знаний разрабатываемой экспертной системы поиска [2]. В математическом смысле выполнение поискового запроса – это одна из форм логического вывода (например, возможность вывести из множества разнородных данных некоторый компактный результат поиска).

Основная задача правил логического вывода базы знаний рассматриваемой экспертной поисковой системы состоит в обработке и структуризации информации, поступающей из сети Интернет. Для этого на этапе анализа Web-документов базу знаний экспертной системы поиска целесообразно разбивать на различные уровни обобщения в рамках исследуемой тематики, достигая при этом выделения узкоспециализированных источников информации. Это позволит получить релевантный результат поиска, основываясь только на определении предметной области сформированного пользователем запроса. Правила вывода можно составить по каждой из тематик, выделенных экспертом для базы знаний разрабатываемой экспертной поисковой системы. При этом фактически под тематикой Web-страницы подразумевается определенный интерес пользователя, указанный им при регистрации в системе.

Предлагаемый вариант организации базы знаний тематических Интернет-ресурсов приведен на рис. 1. Эта база знаний имеет трехуровневую структуру, с которой взаимодействуют соответствующие правила логического вывода. Опишем назначение каждого из уровней:

Уровень 1. Обобщенное понятие – содержит данные, отвечающие за описание основных признаков разделения экспертом источников данных по их тематикам.

Уровень 2. Тематика источника – содержит данные, характеризующие принадлежность Web-ресурса к определенной тематике (при этом вводится ограничение на принадлежность любого Web-ресурса только одной тематике).

Уровень 3. Содержание источника – содержит данные, описывающие структурные характеристики Web-страниц заданного источника информации.

Правила логического вывода участвуют в процессе формирования итогового результата в виде списка Web-страниц, удовлетворяющих запросу пользователя. Сама база знаний экспертной системы поиска основывается на данных, выявленных при анализе Интернет-ресурсов и его семантического описания, составленного согласно стандарту Dublin-core. При этом источники информации поступают из плагина и тематических каталогов в виде ссылок на Web-ресурсы, которые затем обрабатываются с учетом особенностей предложенной структуры базы знаний.

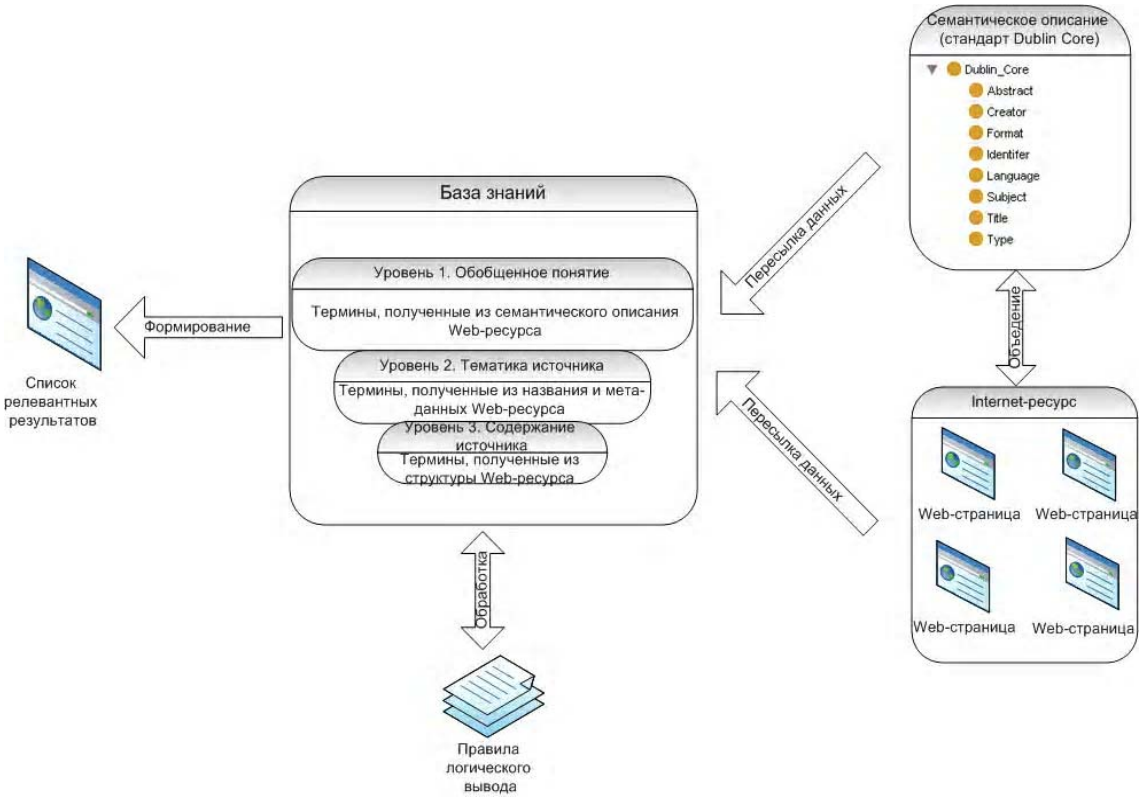


Рис. 1 Структура базы знаний системы

Наполнение базы знаний реализуется с помощью онтологии, которая хранится в виде специального файла с расширением owl, близким по своей структуре к XML-файлу. Под онтологией в данном случае понимаются записанные на особом языке (Ontology Web Language (OWL)) правила и способы описания значений и отношений терминов. Главным ее преимуществом является способность устанавливать синонимию различных терминов. Как только хоть один автор укажет, что два термина являются синонимами, программы-агенты получают возможность конвертировать незнакомые теги (специальные выражения, применяемые для описания онтологии в языке OWL) в известную им систему координат. Это придает дополнительную гибкость технологиям Semantic Web [2], поскольку разработчикам не потребуется обновлять базу знаний после появления новых терминов или онтологий – они всегда смогут конвертировать термины, исходя из их связей с другими понятиями, самостоятельно обучаясь использованию неизвестных ранее тегов.

В базовой версии онтология рассматриваемой модели базы знаний экспертной системы поиска состоит из трех основных терминов: User (Пользователь), Interests (Интересы пользователя) и WebSource (Название страниц, на которые заходил пользователь).

Принцип работы рассматриваемой базы знаний заключается в следующем:

– данные, полученные после регистрации пользователя в экспертной системе поиска, поступают в термины User и Interests. При этом в первом хранится общая информация о нем (ФИО, возраст, email и т.д.), а во втором содержится перечень его интересов (выбранных из предложенного списка при регистрации). Эти термины связаны между собой свойством hasInterests, благодаря которому элементы Interests могут рассматриваться как часть термина User;

– в термин WebSource данные поступают из плагина, который закачивается пользователем при регистрации в экспертной системе поиска (или из тематического каталога). Они представляют собой ссылки на Web-страницы, которые посетил пользователь, а также данные, полученные при анализе их HTML-кода (значимую информацию электронного документа), и атрибут ValuableInformation. Этот атрибут, в свою очередь, состоит из списка элементов, характеризующих его содержание. Термин WebSourse связан с термином User свойством hasVisitWebSourse по аналогии с термином Interests;

– после накопления данных, полученных от пользователей экспертной системы поиска, они могут быть выведены из базы знаний в виде итогового результата в соответствии со сформированным запросом.

Остановимся на термине WebSourse более подробно, поскольку он содержит ключевые объекты базы знаний – Web-документы. Они представляют собой отдельную онтологию, которая имеет заданную экспертом определенную классовую структуру и формируется следующим образом:

– классовая структура онтологии формируется разработчиком на основании проведенных экспериментальных исследований и экспертных данных для термина WebSourse с учетом свойств, определяемых для каждого исследуемого Web-документа;

– экземпляры классов состоят из Web-страниц, которые автоматически классифицируются системой на основании заданных свойств каждого из классов;

– в процессе работы системы для каждого из классов онтологий формируется список ключевых слов, соответствующих их тематике;

для каждой Web-страницы определяется список значений ее структурных характеристик, который интегрируется в сформированную онтологию (рис. 2).

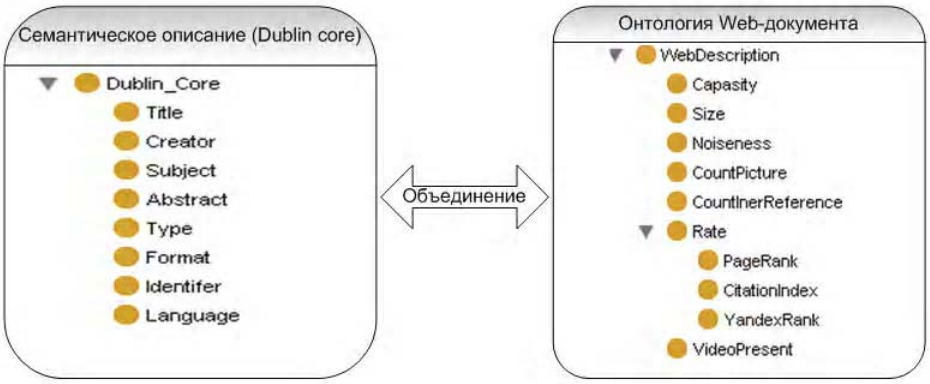


Рис. 2. Онтология Web-документов

Полученная онтология Web-документа объединяется со своим семантическим описанием, составленным в соответствии со стандартом Dublin Core автором Web-документа. В случае его отсутствия генерируется семантическое описание с пустыми значениями.

Сформированная по описанному алгоритму база знаний будет способствовать быстрому и эффективному выводу релевантных результатов поиска в виде списка Web-документов в соответствии с запросом пользователя и с учетом его интересов в рамках заданной предметной области. Это выполняется благодаря использованию современных технологий Semantic Web (OWL, Dublin Core) с помощью предложенной структуры ее организации и построения.

3. Алгоритм поиска релевантной информации с применением социального индексирования Web-документов

В соответствии со своим назначением рассматриваемая экспертная поисковая система должна формировать релевантные результаты в соответствии с текущими требованиями и интересами пользователя. Выполнение этой задачи основано на применении описанных в [2] методов поиска, использующих критерий социального индексирования и оценивание структурных характеристик Web-документа. Рассмотрим подробнее особенности применения этих характеристик поиска в экспертной поисковой системе.

Алгоритм работы метода поиска релевантной информации с применением социального индексирования Web-документов реализуется следующим образом:

– формируются данные об интересах пользователей, полученные на основе заполнения регистрационной формы при работе с экспертной системой поиска;

– накопление данных по разным тематикам выполняется с помощью плагина экспертной системы поиска, который с разрешения пользователя передает информацию о том, какие страницы он посещает и соответствуют ли они его текущим интересам. Полученные данные обрабатываются и передаются в базу знаний;

– пользователь с помощью программы плагина оценивает степень соответствия Web-документа его интересам;

– путем обработки поступающих данных от плагинов пользователей рассчитываются значения социального индекса, которые присваиваются каждому Web-документу;

– на основании полученных значений рассчитывается одна из частей персонализированного рейтинга (вторая часть вычисляется критериями качества информации) популярности Web-документов среди пользователей с близкими интересами, который учитывается при выводе списка релевантных результатов (Web-документы с более высоким рейтингом выводятся в начале списка);

– реализуется проверка на соответствие мета-данных анализируемого Web-документа его реальной тематике (имеет ли он популярность у пользователей с интересами в рамках заданной предметной области). Если источник информации не проходит проверку, экспертная система считает его нерелевантным и не выводит в качестве итогового результата (по желанию пользователя возможен вывод данных источников информации в конце списка);

– на основании социальной значимости каждого Web-документа (его социального индекса) формируется отдельный список наиболее востребованных источников информации по различным тематикам среди пользователей с близкими интересами.

Структурные характеристики – основные критерии оценки качества информации, применяемые разрабатываемой экспертной системой поиска. Значение каждой структурной характеристики Web-документа определяется по средствам интеграции с различными Web-сервисами, а также программного анализа Web-документов. Алгоритм применения данных критериев для вывода релевантных результатов по запросу пользователя реализуется следующим образом:

– пользователь во время формирования поискового запроса к экспертной системе указывает диапазон возможных значений структурных характеристик искомых Web-документов (по умолчанию заданы средние значения);

– на основании полученных значений структурных характеристик Web-документов вычисляется общая оценка качества для каждого из источников информации, которая составляет вторую часть персонализированного рейтинга популярности Web-документов среди пользователей с близкими интересами;

– после того как экспертная система поиска выдаст итоговый результат в виде списка релевантных Web-документов, их порядок может быть изменен путем применения различного типа сортировки по одной или нескольким структурным характеристикам источников информации;

– в случае изменения пользователем диапазона допустимых значений структурных характеристик Web-документов возможен пересчет итоговых результатов.

Таким образом, итоговый результат, соответствующий требованиям и интересам пользователя, формируется экспертной системой в виде персонализированного рейтинга Web-

документов, который зависит от значений социального индекса и общей оценки значимости и определяется по следующей формуле:

$$PR_i = \sqrt{\omega_1 (SI_i^2) + \omega_2 (d(Optimal)^2_i)}, \quad (1)$$

где i – номер анализируемого Web-документа, $i = \overline{1,m}$ (m – общее количество исследуемых Web-документов); PR_i – персонализированный рейтинг i-го Web-документа; SI_i – значение социального индекса i-го Web-документа; $d(Optimal)_i$ – общая оценка значимости i-го Web-документа; $\omega_{1,2}$ – весовые коэффициенты с диапазоном значений, которые устанавливаются пользователем экспертной системы поиска и определяют степень значимости каждого из критериев (по умолчанию имеют равнозначные значения – 0,5 и 0,5 соответственно). Следовательно, пользователь может влиять на итоговый результат работы экспертной системы поиска.

Таким образом, в соответствии с (1) определяется итоговое значение персонализированного рейтинга Web-документа, на основании которого формируется список релевантных результатов экспертной системы поиска в соответствии с запросом пользователя. Это осуществляется путем сравнения значений персонализированного рейтинга Web-документов, которые удовлетворяют запросу пользователя. Чем выше данное значение, тем выше позиция документа в списке релевантных значений.

Опишем основные этапы реализации схемы формирования экспертной системой релевантных результатов по запросам пользователя (с учетом его актуальных интересов):

– заполнение базы знаний Web-страницами из сети Интернет и описание семантических связей в рамках близких предметных тематик в форме онтологий;

– регистрация пользователя поиска с помощью Web-формы в экспертной системе поиска (заполнение идентификационных данных, создание социального профиля и указание своих тематических интересов) и установка плагина, встраиваемого в браузер для оценивания социальной значимости информационного источника;

– загрузка пользователем любой Web-страницы из сети Интернет через браузер, в который встроен плагин экспертной системы поиска;

– формирование критериев социальной значимости Web-страницы на основании оценки пользователя с учетом его интересов и сохранение данных о ней в базе знаний системы;

– вычисление социального индекса Web-страницы (с использованием критериев ее социальной значимости и информации из базы знаний системы);

– определение пользователем весовых коэффициентов и задание возможных значений диапазона структурных характеристик с помощью Web-формы, которая может быть представлена в виде динамического списка, состоящего из элементов описанных выше параметров. Ее реализация возможна на этапе практической разработки модели экспертной системы поиска;

– формирование значений обобщенных критериев качества информации и описание найденных Web-страниц в соответствии с запросом пользователя, представляемых в виде Xml-файлов. При этом анализируемые параметры электронных документов образуются из их семантических описаний (по стандарту Dublin Core), представленных в виде отдельных Rdf-файлов, а также данных, выявленных на этапе заполнения базы знаний;

– определение значений степени близости между найденными Web-страницами и обобщенными критериями качества информации;

– формирование общей оценки качества Web-страниц на основании их удаленности от возможных значений диапазона структурных характеристик источника информации, составленных пользователем экспертной системы;

– вычисление персонализированного рейтинга Web-страницы, зависящего от значений ее социального индекса и критериев качества информации, на основании которого формируется список релевантных результатов поиска разрабатываемой экспертной системы.

Таким образом, экспертная система позволяет пользователю получать ранжированную релевантную информацию из сети Интернет в соответствии с его текущими интересами. Это способствует повышению качества результатов поиска необходимых электронных документов по требуемой предметной области и позволяет сократить временные затраты на их последующую обработку.

4. Формирование адаптивных Web-страниц по результатам семантического поиска

Рассмотрим метод формирования динамических Web-страниц путем выделения значимой информации из Web-ресурсов сети Интернет по запросу пользователя. Данный метод основывается на взаимодействии пользователя с плагином экспертной системы поиска и предусматривает поэтапную реализацию следующих операций:

– поиск необходимого источника информации для формирования персонализированной динамической Web-страницы (выполняется путем формирования пользователем запроса к экспертной системе, а также с помощью других сторонних Web-сервисов или простого серфинга в сети Интернет при условии установки специализированного плагина экспертной системы поиска);

– обработка и выделение значимой информации из Web-документа. Пользователь по средствам плагина дает оценку используемому источнику информации, после чего происходит проверка наличия электронного документа в хранилище значимой информации. В случае положительного результата происходит считывание информации из базы знаний. При ее отсутствии выполняется обработка Web-страницы методом выделения значимой информации с последующей записью в хранилище экспертной системы поиска;

– построение персонализированной адаптивной Web-страницы пользователя. Полученная значимая информация от электронного документа интегрируется в адаптивную Web-страницу. При этом, если у пользователя ее еще нет, то сначала задается структура адаптивной Web-страницы (предлагаемый шаблон приведен на рис. 3) и задается адрес страницы, после чего в нее могут заноситься данные.

При ее отсутствии выполняется обработка Web-страницы методом выделения значимой информации с последующей записью в хранилище экспертной системы поиска [3].

Для определения расстояния между шаблоном и каждым из блоков анализируемого Web-ресурса (с учетом предварительной нормализации величин) используется следующая формула:

$$d_{E_i} = \sqrt{\omega_1 \left(\frac{Tg_i}{Tg_i(max)} \right)^2 + \omega_2 \left(\frac{Sp_i}{Sp_i(max)}\right)^2 + \omega_3 \left(\frac{Sm_i}{Sp_i(max)}\right)^2}, \qquad (2)$$

где i – номер блока анализируемого Web-ресурса; d_{E_i} – евклидово расстояние для i–го блока анализируемого Web-ресурса; $\omega_{1,2,3}$ – весовые коэффициенты для характеристик i–го блока произвольного Web-ресурса; Tg – значение, соответствующее количеству тегов, которые отвечают за форматирование текста; $Tg(max)$ – максимально возможное значение, соответствующее количеству тегов, которые отвечают за форматирование текста среди всех анализируемых Web-ресурсов; Sp – значение, соответствующее количеству специальных символов для вставки текста в Web-страницу; $Sp(max)$ – максимально возможное значение, соответствующее количеству специальных символов для вставки текста в Web- страницу для всех анализируемых Web-ресурсов; Sm – значение, соответствующее количеству произвольных символов в тегах форматирования электронного документа; $Sm(max)$ – максимально возможное значение, соответствующее количеству произвольных символов в тегах форматирования электронного документа для всех анализируемых Web-ресурсов.

Остановимся на описании шаблона адаптивной Web-страницы, синтезируемой на основе интеллектуального анализа информационных ресурсов сети Интернет. От правильности выбора его структуры во многом зависит качество формирования контента в рамках единого персонализированного информационного ресурса с актуальными данными, которые должны генерироваться в соответствии с текущими интересами пользователя.

В соответствии с рис. 3 основными элементами структуры шаблона являются:

– заголовок (содержит имя адаптивной Web-страницы, которое задает пользователь в момент ее создания по средствам плагина экспертной системы поиска);

– список доступных Web-документов (отвечает за вывод поименованного перечня всех источников со значимой информацией, добавленных в персонализированную Web-страницу

в соответствии с актуальными интересами пользователя по средствам плагина экспертной системы поиска);

– значимая информация, полученная от Web-документа (содержит контент выбранного пользователем Web-документа, обработанного с помощью метода извлечения значимой информации;

– служебная информация (корректируется с помощью функциональных кнопок «Обновление», «Дублирование», «Удаление»; «Обновление» – отвечает за обновление информации, хранимой на персонализированной Web-странице, путем ее синхронизации с электронным документом, из которого она была получена.

Заголовок	
Список доступных Web-документов	Значимая информация, полученная от Web-документа
	Служебная информация

Рис. 3. Шаблон персонализированной адаптивной Web-страницы пользователя

Общая схема возможных взаимодействий данного плагина с пользователем в рамках экспертной системы поиска приведена на рис. 4. Стрелки соответствуют рассмотренным выше этапам таких взаимодействий.

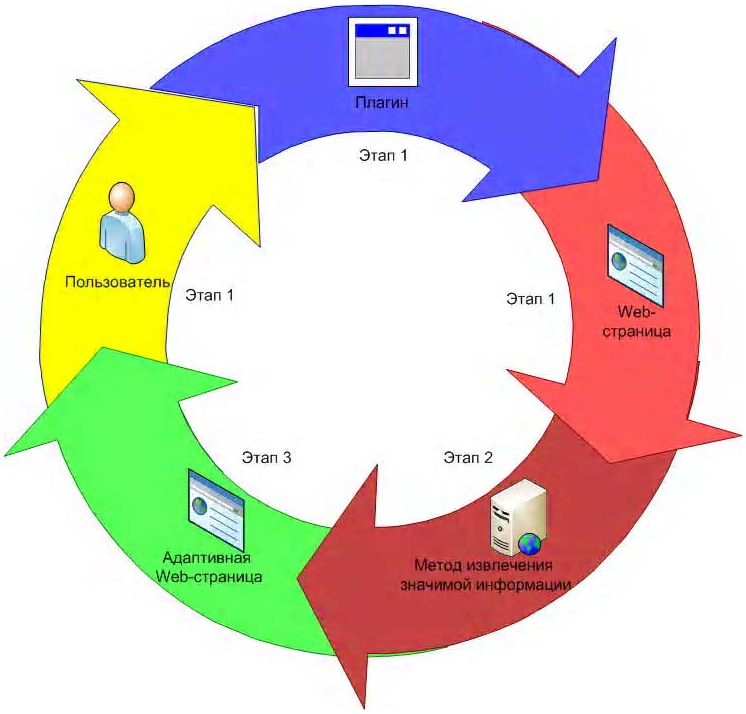


Рис.4. Схема взаимодействия плагина экспертной системы с пользователем

Остановимся подробнее на процессе интеграции значимой информации из Web-ресурса сети Интернет, в состав которого входит электронный документ, представляющий определенный интерес для пользователя, с адаптивной персонализированной Web-страницей, сгенерированной в процессе работы с плагином экспертной системы поиска по описанному выше методу:

– с помощью метода извлечения значимой информации весь контент Web-страницы, отмеченный пользователем с помощью плагина, делится на структурные блоки (рис. 5). При этом в адаптивную Web-страницу попадает только блок со значимой информацией, остальные блоки классифицируются системой как шум;

– индексирование Web-ресурса происходит по средствам анализа его навигационного меню. Для этого на этапе организации хранилища значимой информации данные каждой

страницы Web-ресурса записываются в соответствующий атрибут ValuableInformation термина WebSourse онтологии базы знаний экспертной системы поиска;

–моделирование структуры и содержания адаптивных персонализированных Web-страниц. Формируются порядок и формат вывода полученных Web-данных в результате обработки страниц Web-ресурсов, которые могут представлять интерес для пользователя. Данные берутся из хранилища значимой информации базы знаний экспертной системы поиска.

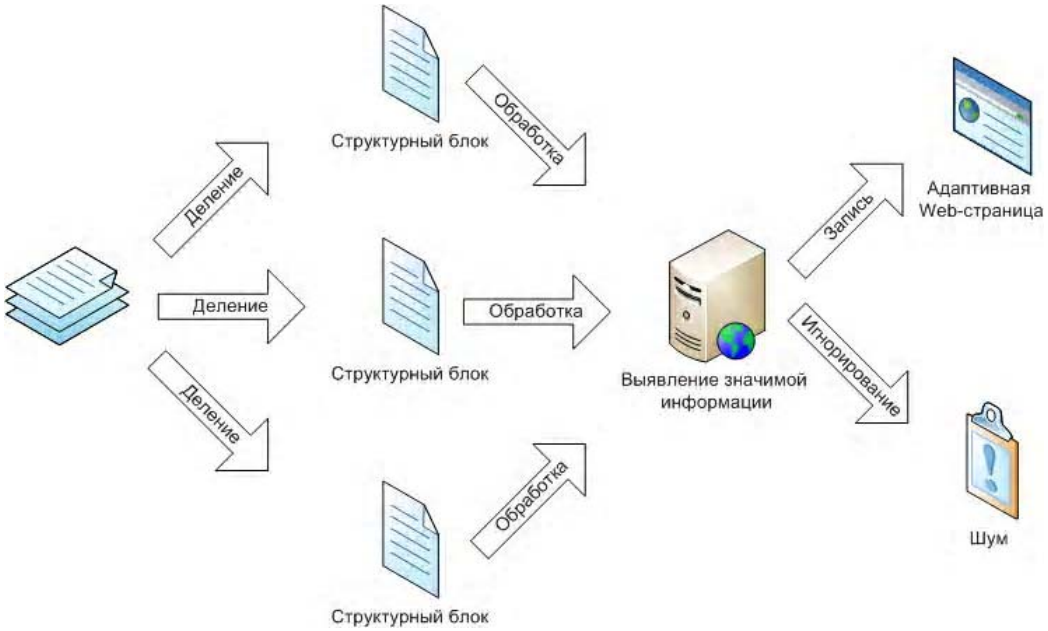


Рис. 5. Схема выделения значимой информации из анализируемых Web-ресурсов

Если все действия были выполнены корректно, то, в конечном счете, пользователь получит разработанный специально для него информационный ресурс с интересующей его информацией в заданном им формате. Это даст ему возможность составлять свой тематический источник информации на основании данных, присутствующих на его персонализированном Web-ресурсе.

Рассмотренная система была реализована в виде поискового программного агента Social Search.

5. Тестирование экспертной поисковой системы

Для оценки качественных характеристик разработанной экспертной системы поиска и синтеза адаптивных Web-страниц на основе Web-ресурса Social Search было выполнено тестирование процедур поиска релевантной информации в сети Интернет для различных типов запросов. Кроме разработанного поискового агента Social Search, в тестовом эксперименте были задействованы поисковые системы Google, Infostream и экспертные системы Similarity/closeness-based resource browser, Digg, Personal Searcher. Результаты эксперимента приведены в таблице.

Анализируемая система	Средняя скорость поиска, с	Наличие собственных критериев оценки	Поиск с учетом семантики запроса
Google Search	1	+	_
Infostream	4	+	_
Similarity/closeness-based resource browser	5	_	+
Personal Searcher *	15	+	_
Digg *	2	+	_
Social Search	3	+	+

Из полученных результатов следует, что экспертная система на основе разработанного программного агента Social Search уступает по скорости работы универсальной поисковой системе Google, но выигрывает у нее в качественном плане, благодаря учету семантики запроса и персонализации релевантного результата, ориентированного на социальную значимость для конечного пользователя. В то же время, по сравнению со специализированными системами (Similarity/closeness-based resource browser, Digg), Social Search показывает, в среднем, такую же скорость поиска, при этом имея более предпочтительную функциональность на этапе фильтрации итогового результата в соответствии с текущими интересами пользователя. Экспертная специализированная система Personal Searcher существенно уступает разработанной экспертной системе по скорости работы и поиску с учетом семантики запроса. В специализированной системе мониторинга медиаресурсов Infostream отсутствует возможность пользователя влиять на позицию Web-страницы в списке релевантных результатов (в отличие от предложенной системы).

6. Выводы

Научная новизна полученных результатов состоит в следующем:

– предложена структура экспертной поисковой системы, которая позволяет не только обеспечивать нахождение необходимых Web-документов с применением семантических методов обработки и социальных критериев, но и обосновывать полученные результаты. Система использует специальный плагин для взаимодействия с пользователями и обмена между ними информацией в соответствии с их интересами;

– предложена схема формирования базы знаний экспертной поисковой системы. Сформированная по такой схеме база знаний будет способствовать быстрому и эффективному выводу релевантных результатов поиска в виде списка Web-документов в соответствии с запросом пользователя и с учетом его интересов в рамках заданной предметной области;

– предложена модель формирования адаптивных Web-страниц, основанная на взаимодействии пользователя с плагином экспертной поисковой системы.

Практическая значимость. Результаты тестирования системы подтверждают возможность и целесообразность ее практического использования при построении адаптивных Web-страниц, учитывающих социальный профиль пользователей.

Перспективным представляется развитие предложенного подхода для создания гибридных систем интеллектуального анализа информационных ресурсов сети Интернет.

Список литературы: 1. *Джарратано Д., Райли Д.* Экспертные системы: принципы разработки и программирование / М.: ООО "И.Д. Вильямс", 2007. 1152 с. **2.** *Почанский О.М.* Социальное индексирование Web-документов для семантического поиска // Искусственный интеллект. 2012. №1. С. 112-122. 3. *Гаврилова Т.А., Хорошевский В.Ф.* Базы знаний интеллектуальных систем / Спб: Питер, 2000. 384 с.

Поступила в редколлегию 12.06.2012

Почанский Олег Михайлович, аспирант кафедры искусственного интеллекта ХНУРЭ. Научные интересы: методы искусственного интеллекта, семантический поиск в системах интеллектуальной обработки данных. Адрес: Украина, 61166, Харьков, пр. Ленина, 14.

УДК 519.7:007.52

І.М. ГАЛУШКА, В.В. ЗАВГОРОДНІЙ, С.М. СОЛОШИЧ, С.С. ЩЕРБАК

УДОСКОНАЛЕННЯ ТЕХНОЛОГІЙ ДОСТУПУ ТА ОБРОБКИ ПОВ'ЯЗАНИХ ДАНИХ СЕМАНТИЧНИХ ДОДАТКІВ LINKEDDATA

Зростання популярності концепції пов'язаних даних поряд з описом інформації у вигляді триплетів RDF зумовило необхідність дослідження процедур взаємодії зі сховищами триплетів. У результаті розроблено універсальний пошуковий інтерфейс, що забезпечує візуальну побудову запитів до сховищ триплетів. Візуальні запити автоматично перетворюються у мову запитів SPARQL, яка використовується для доступу до сховищ триплетів. Після виконання запиту користувач отримує контекст з триплетами, що відповідають шуканим з урахуванням заданих обмежень на предикати і об'єкти.

Вступ

Широкий розвиток стандартів і підтримка концепції LinkedData великими IT компаніями визначає тенденції розвитку майбутнього WWW як глобальної бази даних, в якій можна через спеціалізовані пошукові інтерфейси отримувати доступ до структурованих представлень даних, документів, розподілених по Web, для вирішення завдань пошуку і подібних.

В рамках LinkedData інформація описується в термінах мови RDF (англ. ResourceDescription Framework), а саме, у вигляді триплетів, трійок вигляду «суб'єкт-предикат-об'єкт» або квад (quad) - іменованих графів виду «граф-суб'єкт-предикат- об'єкт». Далі, якщо це не буде призводити до протиріччя, будемо використовувати поняття «триплет» для поняття як «триплет», так і «квад».

Модель даних RDF передбачає розподілене зберігання об'єктів і їх схем, за їх наявності, на різних web-серверах з інтегрованим або зовнішнім спеціалізованим сховищем триплетів (Triplestore). Для доступу до сховищ триплетів використовується протокол і мова запитів SPARQL (англ. SPARQL Protocoland RDF QueryLanguage), яка є в певному сенсі аналогом мови SQL, що використовується для обробки даних реляційних баз. Використання подібних мов запитів передбачає наявність спеціалізованих знань для ефективного їх застосування, що не сприяє ні популяризації мови SPARQL як засобу отримання даних, ні збільшенню поширеності сховищ триплетів.

Метою даної роботи є підвищення ефективності пошуку на основі сховищ триплетів шляхом зменшення складності процедур динамічного формування запитів на мові SPARQL і застосування шаблонів запитів для генерації пошукового інтерфейсу користувача. Тому до задач дослідження відносяться:

– аналіз підходів щодо використання та дослідження можливостей пов'язаних даних в контексті пошуку;

– формалізація пов'язаних даних та процедур пошуку;

– дослідження технологій побудови сучасних інтерфейсів користувача для сховищ пов'язаних даних;

– розробка архітектури та реалізація типового рішення пошукового інтерфейсу користувача на базі мови запитів SPARQL.

1. Пошукові інтерфейси користувача до сховищ триплетів

Пошук інформації в рамках LinkedData ґрунтується на використанні даних, представлених у вигляді об'єктів, що належать до деякої предметної області. У рамках такого підходу вміст документа подається як сукупність об'єктів, об'єднаних деяким контекстом. Для визначення контексту об'єктів будемо використовувати поняття «граф» квада. Тоді пошук інформації зводиться до ідентифікації за заданими критеріями (користувальницькими обмеженнями) триплетів графа і висновку контекстів з триплетами, що відповідають ідентифікованим.

Для створення ефективних пошукових інтерфейсів, «дружніх» до користувача, необхідна розробка типових рішень – патернів реалізацій, які могли б дозволити візуально конструювати запити до іменованих графів сховища триплетів і не вимагали б від розробника або користувача програмних систем спеціалізованих знань про мову SPARQL.

На прикладі веб-додатку розглянемо архітектуру класичного рішення на основі сховища триплетів і мови запитів SPARQL (рис.1).

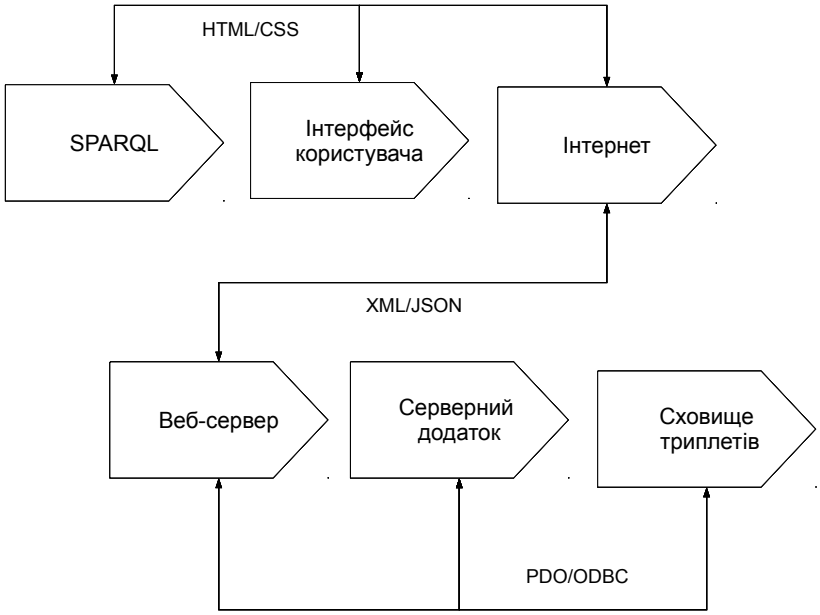


Рис. 1. Архітектура веб-додатку на основі сховища триплетів

Запит користувача на мові SPARQL передається від web-додатку, клієнтська частина якого написана на мовах HTML та CSS, до сховища триплетів, де відбувається виконання запиту через інтерфейси доступу PDO або ODBC, а результати відсилаються назад клієнту в форматі XML або JSON.

Ключова особливість класичного рішення полягає в необхідності формування, найчастіше вручну, запиту на мові SPARQL, що вимагає спеціалізованих знань не тільки про цю мову, але і знань про структуру об'єктів, що містяться у сховищі триплетів.

Одним з додатків, що реалізують подібну функціональність, є веб-інтерфейс isql компанії OpenLink, що застосовано в Dbpedia, який поряд з універсальністю доступу до даних відрізняється стабільністю роботи.

Розглянемо архітектуру запропонованого рішення, що виключає взаємодію користувача з сховищем триплетів через SPARQL (рис.2).

Рис. 2. Архітектура візуального засобу побудови запитів SPARQL до сховища триплетів

У рамках такого рішення користувач через веб-інтерфейс візуального засобу побудови запитів SPARQL отримує доступ до переліку іменованих графів сховища триплетів, вибір з яких дасть можливість автоматично отримати інформацію про топологію графа, а саме про його предикати і типи даних об'єктів, що може бути використано для формування пошукового запиту шляхом встановлення користувальницьких обмежень (фільтрів) на дані, які виводяться з сховища триплетів.

Обмеження користувача визначимо як набір правил з умовами виведення об'єктів іменованого графа, типи яких визначають семантику операцій порівняння, що застосовуються. Як основу для умов виведення визначимо перелік операцій, а саме «=» (дорівнює), «>» (більше), «<» (менше), які можуть бути застосовані до об'єктів.

Семантика операцій «=», «>», «<» визначається типом об'єктів іменованого графа, тобто яким саме чином буде реагувати конструктор запитів на символьні імена операцій «=» «>» «<».

Для цілочисельного типу об'єкта (xsd: integer) операції «=», «>», «<» мають такий сенс:

«=» – порівнювані об'єкти рівні;

«>» – об'єкт, з яким проводиться порівняння, більший порівнюваного;

«<» – об'єкт, з яким проводиться порівняння, менший порівнюваного.

Для рядкового типу (xsd: string) під операціями «=», «>», «<» будемо розуміти таке:

«=» – порівнювані об'єкти рівні;

«>» – порівнюваний об'єкт є підрядком об'єкта, з яким проводиться порівняння;

«<» – об'єкт, з яким проводиться порівняння, є підрядком порівнюваного.

Дії «=», «>», «<» для об'єктів типу «xsd: float» (число з плаваючою точкою) мають таку ж семантику, як і відповідні операції з цілочисловим типом об'єктів.

Таким чином, користувач може формувати правила виводу вмісту того чи іншого іменованого графа сховища триплетів. На основі отриманих правил може бути автоматично сформований типовий SPARQL-запит до сховища триплетів, практичні аспекти створення якого розглянуті в розділі 3, а відповідна формальна модель триплетів і пов'язаних даних в цілому представлена в розділі 2.

2. Формальна модель пов'язаних даних на основі частковизначених схем

Пов'язані дані, як елемент концепції LinkedData, формально можуть бути представлені за допомогою формули:

$$t =< g, s, p, o >, \quad (1)$$

де t – триплет; g – іменований граф; s – суб'єкт; p – предикат; o – об'єкт.

Сукупність структур t, визначених формулою (1), будемо вважати сховищем даних:

$$T = \{t_i\}, i = \overline{(1,n)}, \quad (2)$$

тут t_i – i-й триплет; n – кількість триплетів у сховищі.

Враховуючи необхідність використовувати для пошуку інформацію про контекст пов'язаних даних, будемо вважати, що всі t_i, в яких g однакове, об'єднані одним контекстом. Контекст визначимо такою формулою:

$$G = \{g_j\}, j = \overline{(1,m)}, \quad (3)$$

де G – множина контекстів сховища; g_j – j-контекст сховища даних; m – кількість контекстів сховища.

Кожному контексту, згідно з формулою (1), поставимо у відповідність трьохелементний набір <s,p,o>. Таким чином, контекст визначимо за допомогою формули:

$$\forall g \in G : g =< S, P, O >, \quad (4)$$

тут g – контекст сховища даних; G – множина всіх контекстів; S – множина суб'єктів; P – множина предикатів; O – множина об'єктів.

Розширимо поняття об'єкта так:

$$\forall o \in O : o =< T, L, V >, \quad (5)$$

де T – тип даних; L – мова представлення; V – значення.

Враховуючи особливість концепції LinkedData, а конкретно, необов'язковість визначення схем, типів даних та мов, на яких представлені значення об'єкта O у випадку рядкового типу, будемо вважати елементи об'єкта o необов'язковими для визначення, а схему об'єкта, згідно з формулою (5) будемо вважати частковозначеною схемою пов'язаних даних, які визначаються формулами (1)-(4).

Враховуючи, що на множині G необхідно вирішувати пошукові задачі, модифікуємо формулу (4) шляхом введення допоміжного елемента - множини функцій, які можуть виконуватись на множинах елементів контекстів G:

$$\forall g \in G : g =< S, P, O, F >, \quad (6)$$

де g – контекст сховища даних; G – множина всіх контекстів; S – множина суб'єктів; P – множина предикатів; O – множина об'єктів; F – множина функцій.

Практичні аспекти реалізації функцій F формули (6) покладемо на сторонніх розробників програмного забезпечення та наведемо приклади запитів на мові SPARQL, на основі яких можуть бути реалізовані функції F, в наступному розділі.

Зауваження. Далі, якщо це не буде призводити до протиріччя, «контекст», «іменований граф» будемо використовувати як синоніми.

3. Практичні аспекти реалізації пошукового інтерфейсу до сховища триплетів

Формальна модель пов'язаних даних з частковозначеними схемами (наведена на рис. 2) та архітектура веб-інтерфейсу візуального засобу побудови запитів SPARQL до сховища триплетів передбачають виконання набору зумовлених дій за допомогою SPARQL для динамічного отримання топології іменованого графа. До таких дій відносяться:

• отримання списку іменованих графів;
• отримання списку предикатів іменованого графа;
• отримання типу даних предиката.

Для отримання списку всіх наявних іменованих графів у сховищі триплетів можна використовувати такий запит на мові SPARQL:

```
SELECT distinct ?G WHERE
{
GRAPH ?G {?S ?P ?O}
}
```

Для отримання конкретного іменованого графа або групи, ім'я яких відповідає деякому критерію, можна використовувати такий запит на мові SPARQL:

```
SELECT distinct ?G WHERE
{
GRAPH ?G
    {?S ?P ?O. Filter (regex (?G, <http://shcherbak.net/>))
    }
}
```

де з переліку всіх графів обираються ті, у яких http://shcherbak.net/ зустрічається в імені графа.

Для отримання списку предикатів в іменованому графі можна використовувати такий запит на мові SPARQL:

```
SELECT distinct ?P
FROM <http://shcherbak.net/User>
{
?S ?P ?O
}
```

де з графа <http://shcherbak.net/User> будуть вибрані всі унікальні предикати, які повернуті у вигляді набору як результат вибірки.

Для отримання типу предиката можна використовувати такий запит на мові SPARQL:

```
SELECT datatype (?O)
FROM <http://shcherbak.net/User>
WHERE
```

```
{
?S ns: date_of_ birthday ?O
}
```

де для предиката ns:date_of_birthday графа http://shcherbak.net/User буде отриманий тип об'єкта.

На основі подібних запитів можна будувати довільні запити на вибірку з іменованих графів, наприклад:

```
sparql SELECT? s? p? o
FROM <http://shcherbak.net/User>
WHERE
{? S? P? O
  FILTER (? S = ns: 2)
}
```

де з графа http://shcherbak.net/User будуть обрані триплети користувача ns:2.

Крім того, як обмеження користувача можуть виступати мовні теги (langtags). У цьому випадку, для виведення інформації з графа http://shcherbak.net/User можна встановити фільтр на вибірку об'єктів російською мовою, у яких мовний тег встановлено у значення «ru»:

```
SELECT? Name? Second_name
FROM <http://shcherbak.net/User>
WHERE
{? S ns: name? Name.
  ? S ns: second_name? Second_name.
  FILTER (? Name = 'Іван' @ ru&&? Second_name = 'Іванов' @ ru)
}
```

Для пошуку об'єкта з урахуванням декількох користувальницьких обмежень можна використовувати такий SPARQL-запит:

```
PREFIX ns: <http://shcherbak.net/>
SELECT? O
FROM <http://shcherbak.net/User>
WHERE
{? S ns: name? O
   FILTER regex (? O, "Іван", "i")
   FILTER regex (? O, "Петров", "i")
 }
```

Реалізація пошукового інтерфейсу на основі наведених вище запитів (рис.3) надає користувачу допоміжну інформацію, яка може значно полегшити процедуру складання запиту до сховища триплетів і зменшити час його складання.

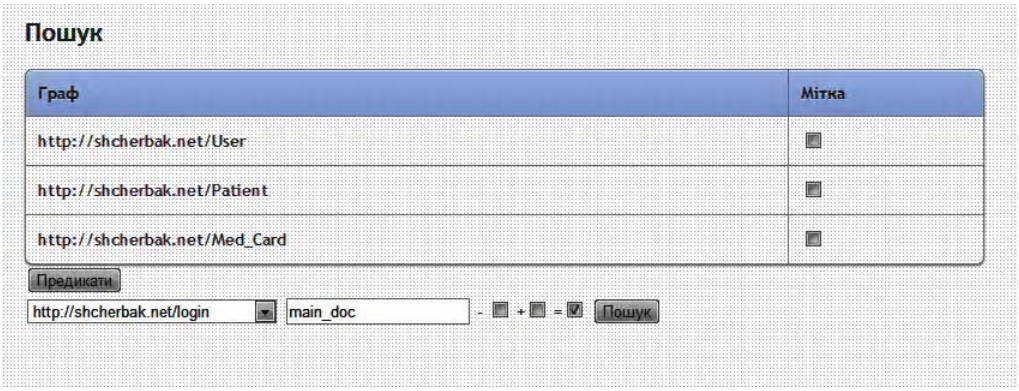


Рис. 3. Приклад пошукового інтерфейсу

Процедура складання запиту зводиться до вибору іменованих графів, що представляють зацікавленість в контексті пошуку, і накладення обмежень на предикати та значення виводяться в результаті пошуку об'єктів.

Результати пошуку будуть згруповані по приналежності до деякого суб'єкта в табличному вигляді (рис. 4). Таким чином, на стороні клієнта з множини триплетів, отриманих у результаті виконання запиту SPARQL на стороні сервера, компонується запис у зручному для читання вигляді.

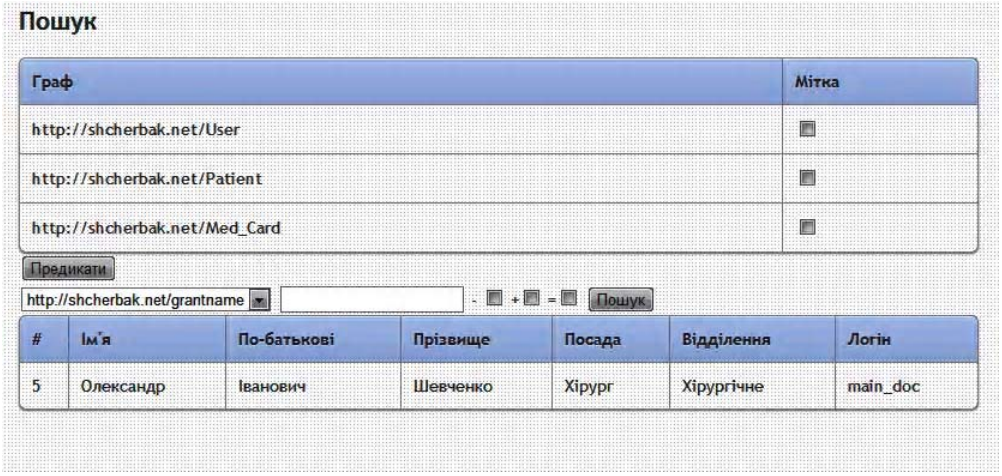


Рис. 4. Приклад виведення результатів пошуку в табличному поданні

Мова RDF дозволяє формувати довільної складності графові структури для представлення даних, що складає як в обчислювальному плані, так і в плані ефективного використання проблеми по організації взаємодії через інтерфейс ODBC (англ. Open Database Connectivity) зі сховищем триплетів, тому розглянемо обмеження запропонованого рішення.

Передбачається, що дані, які містяться у сховищі триплетів, структуровані з використанням принципів об'єктно-орієнтованого проектування, а саме що іменований граф є контейнером (класом) для множини об'єктів цього класу, однозначно ідентифікованих по суб'єкту триплета.

Обмеження дають можливість більш зручної роботи зі сховищем триплетів для розробників програмного забезпечення, що використовують у своїй роботі реляційні системи управління базами даних. Без врахування обмежень запропоноване рішення є працездатним, але реальний сенс генерації даних як результатів пошуку зміниться.

Для врахування обмежень на дані об'єкти пропонується створювати іменовані графи на основі типових, запропонованих нижче, SPARQL запитів.

Для створення іменованих графів можна використовувати такий SPARQL-запит:

CREATE GRAPH <http://shcherbak.net/User>

де створюється граф користувача «http://shcherbak.net/User» в поточному сховищі триплетів.

Для додавання триплетів в іменований граф можна використовувати наступний SPARQL-запит:

```
INSERT DATA INTO
<http://shcherbak.net/User>
{
  ns: log1 ns: login “user1”.
  ns: log1 ns: firstname “Іван”.
  ns: log1 ns: lastname “Іванович”.
  ns: log1 ns: grantname “Іванов”.
}
```

Висновки

1. Запропоновано архітектуру та наведено приклад реалізації візуальних засобів побудови запитів SPARQL, орієнтована на підвищення швидкості введення запитів та покращення якості пошуку в сховищах даних, який дозволяє на основі частковизначених схем об'єктів проводити пошук даних.

2. Запропонована формальна модель пов'язаних даних на основі частково-визначених схем.

3. Отримала подальший розвиток модель процесу пошуку даних в розподілених середовищах на основі частковоівизначених схем.

4. Запропоновані технологічні рекомендації щодо реалізації інтерфейсів користувача до сховищ триплетів та їх автоматичного формування.

5. Промислова значущість запропонованих моделей та технологій полягає у високій ринковій привабливості та зацікавленості компаній в новітніх рішеннях щодо підвищення ефективності пошукових засобів семантичних додатків.

6. Запропонована архітектура універсального пошукового інтерфейсу, яка забезпечує візуальну побудову та виконання запитів до сховищ триплетів на мові SPARQL. Після виконання запиту користувач отримує контекст з триплетами, що відповідають потрібним з урахуванням заданих обмежень на предикати і об'єкти.

7. Практична реалізація рішення створена на мові програмування PHP і пройшла успішну апробацію на базі серверу OpenLinkVirtuoso.

Список літератури: 1. *DuCbarme B.* Learning SPARQL / B.DuCbarme. O'ReillyMedia: 2011. 258 с. **2.***Powers S.* Practical RDF / S. Powes. O'ReillyMedia: 2008. 352 с. **3.** *Бек К.* Шаблоны реализации корпоративных приложений.: Пер. с англ. М.:ООО "И. Д. Вильямс", 2008. 176 с.

Надійшла до редколегії 12.06.2012

Галушка Ілона Миколаївна, асистент кафедри ІУС КрНУ. Наукові інтереси: інтелектуальні алгоритми, семантичні додатки, інтеграційні процеси. Адреса: Україна, 39600, Кременчук, вул. Першотравнева, 20 , тел./факс: (05366) 3-60-00. E-mail: anoli@gmail.com.

Завгородній Валерій Вікторович, старший викладач кафедри ПЗС ДДТУ. Наукові інтереси: інтелектуальні алгоритми, бази даних. Адреса: Україна, Дніпродзержинськ, вул.Дніпробудівська, 2, тел. (0569) 55-13-89.E-mail: valera_ddtu@i.ua.

Солошич Сергій Миколайович, аспірант кафедри ІУС КрНУ. Наукові інтереси: інтелектуальні алгоритми. Адреса: Україна, 39600, Кременчук, вул. Першотравнева, 20 , тел./факс: (05366) 3-60-00. E-mail: soloshich@gmail.com.

Щербак Сергій Сергійович, доцент кафедри ІУС КрНУ, канд. техн. наук, доцент, с.н.с. Наукові інтереси: інтелектуальні алгоритми, семантичні додатки, агентні технології. Адреса: Україна, 39600, Кременчук, вул. Першотравнева, 20, тел./факс: (05366) 3-60-00.E-mail: sergey.shcherbak@gmail.com. www: http://щербак.net.

УДК 681.5+548.55

А.П. ОКСАНИЧ, С.Э. ПРИТЧИН, В.В. МАЛЁВАНЫЙ

РАЗРАБОТКА МАТЕМАТИЧЕСКОЙ МОДЕЛИ И УСТРОЙСТВА АВТОМАТИЧЕСКОГО КОНТРОЛЯ И ПОДДЕРЖАНИЯ ДИАМЕТРА СЛИТКОВ ГЕРМАНИЯ ВЫРАЩИВАЕМЫХ ПО МЕТОДУ ЧОХРАЛЬСКОГО

Предлагается математическая модель и схема теплового узла установки выращивания слитков германия, включающая в себя систему сервоконтроля с замкнутой петлей обратной связи, в основу которой положен метод взвешивания выращиваемого слитка германия. Для решения задачи автоматизации процесса выращивания предлагается функциональная схема устройства автоматического контроля и поддержания диаметра выращиваемого слитка германия. Приводятся решения, которые позволят существенно сократить потери монокристалла германия при механической калибровке слитков, а также повысить их структурные характеристики.

1. Введение

Высокопрозрачные в ближней ИК-области монокристаллы германия широко применяются в инфракрасной оптике. Однако возникающие при выращивании монокристаллов трехмерные дефекты кристаллической решетки – пузырьки, поры, включения посторон-

них фаз, в силу своей соизмеримости с длинами волн света являются также и оптическими дефектами, влияющими на качество пропускания света. Совокупность дефектов меньших размерностей – атомов примесей, вакансий дислокаций приводит к локальным неоднородностям диэлектрической проницаемости и является источниками различных аномалий, влияющих в большой степени на основной показатель оптического качества монокристаллического германия – коэффициента поглощения [1].

Кроме того, развитие фотоэнергетики, изготовление солнечных элементов на основе гетероструктур привело ко «второму рождению» технологии германия, бывшего когда-то первым «классическим» материалом в технике полупроводников и вытесненного затем кремнием. Стоимость германия как подложечного материала ниже, чем используемого для этого арсенида галлия, не говоря уже о его технологических достоинствах (механическая устойчивость при постростовой обработке) и возможности быть включенным в процесс фотоэлектрического преобразования в многокаскадной гетероструктуре.

Однако выращивание монокристаллов германия большого диаметра (больше 100 мм), предназначенного для промышленного использования в качестве подложечного материала, а также для оптического применения, сопряжено с большими материальными затратами. В связи с этим особенно актуальной становится проблема повышения структурного качества выращиваемых слитков германия путем совершенствования систем управления.

Одним из основных параметров в процессе роста слитков германия является его диаметр. Обеспечение постоянства диаметра растущего слитка по всей длине его цилиндрической части – самая актуальная задача управления процессом выращивания. Колебания диаметра во время выращивания приводят к структурному несовершенству кристаллов, появлению пластической деформации и другим неоднородностям кристаллической решетки, что сказывается на оптическом качестве производимого материала.

Поэтому разработка математической модели и устройства автоматического контроля и поддержания диаметра слитков германия, выращиваемых по методу Чохральского, и является целью данной работы.

2. Постановка задачи

Большинство полупроводниковых монокристаллов производится в промышленных условиях методом Чохральского, т.е. вытягиванием из расплава. Во время процесса выращивания контролируется уровень мениска, образованный капиллярами между разделом границы и основной частью расплава. Необходимость в этом определяется двумя факторами:

1) форма мениска значительно влияет на тепловой поток в центральной части, которая в свою очередь определяет расположение уровня кристаллизации;

2) увеличение поперечного сечения кристалла вызвано поведением уровня мениска.

Установлено [2,3], что процесс стабилизации параметров в условиях контроля диаметра кристалла при выращивании слитков германия по методу Чохральского дает хорошие результаты. Успешная регулировка диаметра зависит от оперативного решения задачи автоматического контроля, т.е. качества сигнала датчика взвешивания и качества обратной связи.

В настоящее время метод взвешивания является наиболее применяемым в автоматических системах контроля диаметра слитка в установках выращивания методом Чохральского. По уровню изменения в весе слитка можно получить сигнал, показывающий различие в диаметре кристалла. Этот сигнал обычно используется для корректировки термических условий и скоростей роста в установке выращивания слитков, чтобы создать систему сервоконтроля с замкнутой петлей. Поэтому для построения автоматизированной системы контроля и поддержания диаметра слитка германия необходимо найти эти условия.

3. Разработка математической модели автоматизированной системы контроля

Рассмотрим схему теплового узла выращивания слитка германия, представленную на рис. 1.

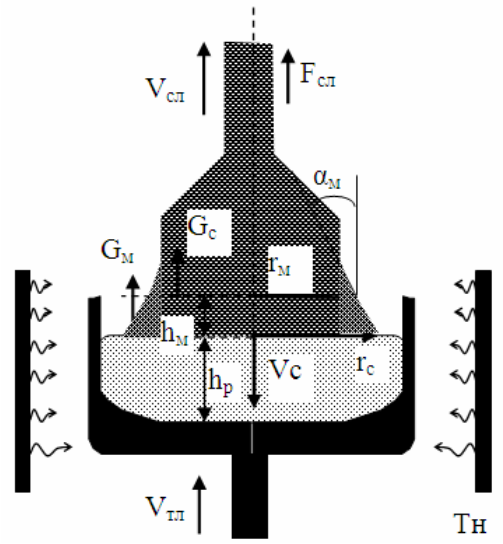


Рис. 1. Схема теплового узла для выращивания слитка германия

Как известно, при выращивании слитков методом Чохральского столбик расплава, осуществляющий связь растущего кристалла с расплавом, поддерживается силой поверхностного натяжения и формирует мениск между поверхностью расплава и растущим кристаллом. При этом граница расплав-кристалл (т. е. фронт кристаллизации) оказывается расположенной над поверхностью расплава. Мениск формируется капиллярными силами и имеет высоту $h_м$ диаметром $r_м$. Информацию о радиусе слитка r_c можно получить из параметров мениска. При выращивании слитков германия применение непрямых методов контроля параметра мениска, например оптических, затруднительно. Применение весового метода, как показано в работах [4,5], позволяет определить угол между расплавом и слитком (угол наклона мениска) $\alpha_м$ и скорость роста $V_к$. Основными регулирующими параметрами, которые влияют на радиус (диаметр) слитка, являются мощность нагревателя Тн и скорость роста $V_к$, представляющая собой разницу скоростей подъема слитка $V_{сл}$ и подъема тигля $V_{тл}$, $V_к = (V_{сл} - V_{тл})$. Два других параметра – частота вращения тигля $\omega_{тл}$ и частота вращения слитка $\omega_{сл}$ в свою очередь тоже влияют на распределение температурных полей в тигле и, следовательно, на диаметр слитка, но их влияние незначительно по сравнению с названными выше.

Взяв за основу работу [6], можно представить зависимость изменения радиуса слитка германия как:

$$r_c^{/} = Vк \times tg(\alpha_м), \tag{1}$$

где

$$\alpha_м^{/} = \frac{V_{сл} - V_{тл} - V_к \times C_{\alpha z}(r_м, \alpha_м)}{C_{\alpha n}(r_м, \alpha_м)}, \tag{2}$$

вычисление коэффициентов $C_{\alpha z}$ и $C_{\alpha n}$ показано в [5].

В свою очередь скорость роста слитка $V_к$ определяется скоростью его выращивания, скоростью подъема тигля, изменением уровня расплава $h_р^{/}$ и изменением высоты мениска $h_м^{/}$ в соответствии со следующим выражением:

$$V_к = V_{сл} - V_{тл} - h'_р - h'_м = \frac{V_{сл} - V_{тл}}{C_{\alpha z}(r_м, \alpha_м) + C_{\alpha n}(r_м, \alpha_м)\dfrac{d\alpha_m}{dt}}. \tag{3}$$

Главным условием, от которого зависит скорость роста, является нагрев зоны роста. Тепловой баланс системы расплав – слиток определится как:

$$Q_c^{/} = Q_p^{/} + Q_к^{/}, \quad (4)$$

где $Q_c^{/}$ – тепловой поток, проходящий через слиток; Q_p – тепловой поток, проходящий через расплав; $Q_к^{/}$ – тепловой поток кристаллизации. Тепловые потоки можно выразить как:

$$Q_c^{/} = \lambda_c S G_c, \quad (5)$$

$$Q_p^{/} = \lambda_p S G_p, \quad (6)$$

$$Q_к^{/} = \rho_c S \Delta K_c V_c. \quad (7)$$

Следовательно, скорость роста определится выражением:

$$V_c = \frac{\lambda_c G_c - \lambda_p G_p}{\rho_c \Delta K_c}, \quad (8)$$

где $\lambda_c = 17\ Wm^{-1}K^{-1}$ – коэффициент теплопроводности германия; λ_p – коэффициент теплопроводности расплава германия; S – площадь поперечного сечения границы расплав-слиток; ρ_c – плотность германия = 5550 кг/м, ΔK_c = 460 кДж/кг – скрытая теплота кристаллизации; G_c – температурный градиент в слитке; G_p – температурный градиент в расплаве.

Максимальная скорость роста ограничивается максимальной величиной скрытой теплоты кристаллизации. Таким образом, общая динамика системы в основном определяется свойствами твердой фазы, т.е. отношением $\frac{\lambda_c G_c}{\rho_c \Delta K_c}$. В то же время произведение $\lambda_p G_p$ является потоком тепла, направленного от расплава в зону роста, и определяет влияние управляющих воздействий на рост слитка. Тогда можно записать выражение для установления температурного градиента:

$$G_p = \frac{(T_p - T_п)}{h_p}, \quad (9)$$

где T_p – температура расплава; $T_п$ =937,5 °C – температура плавления германия; h_p – высота мениска. Значение G_p, равное 3,5 K/mm для германия диаметром 50 мм, получено в работе [7].

Скорость вытягивания определяется выражением:

$$V_c = \frac{\lambda_c}{\rho_c L}\left(\frac{\partial T_c}{\partial z}\right), \quad (10)$$

здесь $L = 4{,}1 \cdot 10^5$ Дж/кг – удельная теплота плавления германия; $\frac{\partial T_c}{\partial z}$ – осевой градиент температуры

Также данный параметр можно определить как зависимость величины прироста высоты слитка ∂z в единицу времени d_t:

$$V_c = \frac{\partial z}{\partial t}. \quad (11)$$

Определим зависимость диаметра слитка через массу слитка:

$$D_c^2 = \frac{4}{\pi\rho}\left(\frac{\partial m_c}{\partial z}\right), \quad (12)$$

где D_c – диаметр слитка; $\partial\, m_c$ – прирост массы слитка, измеренной датчиком веса; ρ_c – плотность германия в твердом состоянии; ∂z – высота прироста слитка, которая вычисляется с помощью скорости вытягивания через промежутки времени, равные 10 с.

Приравняв (10) и (11) получим следующее соотношение:

$$\frac{\partial z}{\partial t} = \frac{\lambda_c}{\rho_c L}(G_c). \tag{13}$$

Далее определив из (13) значение ∂z и подставив его в (12), получим зависимость изменения диаметра кристалла от температуры:

$$D_c^2 = \frac{4L}{\pi\lambda_c}\frac{1}{G_c}\frac{\partial m_c}{\partial z}. \tag{14}$$

Алгоритм управления выращиванием кристалла формируется при выполнении условий постоянства площади поперечного сечения (S_c) и линейной скорости (V_c) роста слитка германия:

$$\frac{dV_c}{dt} = 0, \tag{15}$$

$$\frac{dS_c}{dt} = 0. \tag{16}$$

И если массоперенос в системе осуществляется только на фронте кристаллизации, то можно записать :

$$\frac{dm_p}{dt} = \frac{dm_c}{dt}, \tag{17}$$

где m_p и m_c –массы расплава и слитка, соответственно.

При условии постоянства формы ФК обеспечение (15) и (16) предполагает выполнение условия постоянства расхода расплава:

$$\frac{d^2 m_p}{dt^2} = \frac{d^2 m_c}{dt^2} = 0. \tag{18}$$

Условие (17) является необходимым, но не достаточным. В качестве дополнительного требования должно выполняться условие постоянства S_c или V_c. Последнее возможно при постоянстве скорости вытягивания слитка $V_{сл}$, диаметра тигля $D_{тл}$.

4. Разработка функциональной схемы устройства автоматического контроля и поддержания диаметра слитка германия

На рис. 2 представлена функциональная схема устройства автоматического контроля и поддержания диаметра выращиваемого слитка германия.

При условии постоянства диаметра тигля и скорости вытягивания кристалла управление величиной поперечного сечения слитка или его диаметра может осуществляться изменением конфигурации температурного поля в системе путем изменения температуры расплава.

Поэтому при $V_{сл} = \text{const}$ и $D_{тл} = \text{const}$ для управления процессом выращивания кристаллов путем корректировки температуры нагревателей необходимо наличие текущей информации о следующих величинах:

– скорость подьема слитка – определяется блоком БУПВС;

– диаметр кристалла – рассчитывается ЭВМ на основании данных, получаемых из блока БИВС;

– температура нагревателя, получаемая из блока БПП.

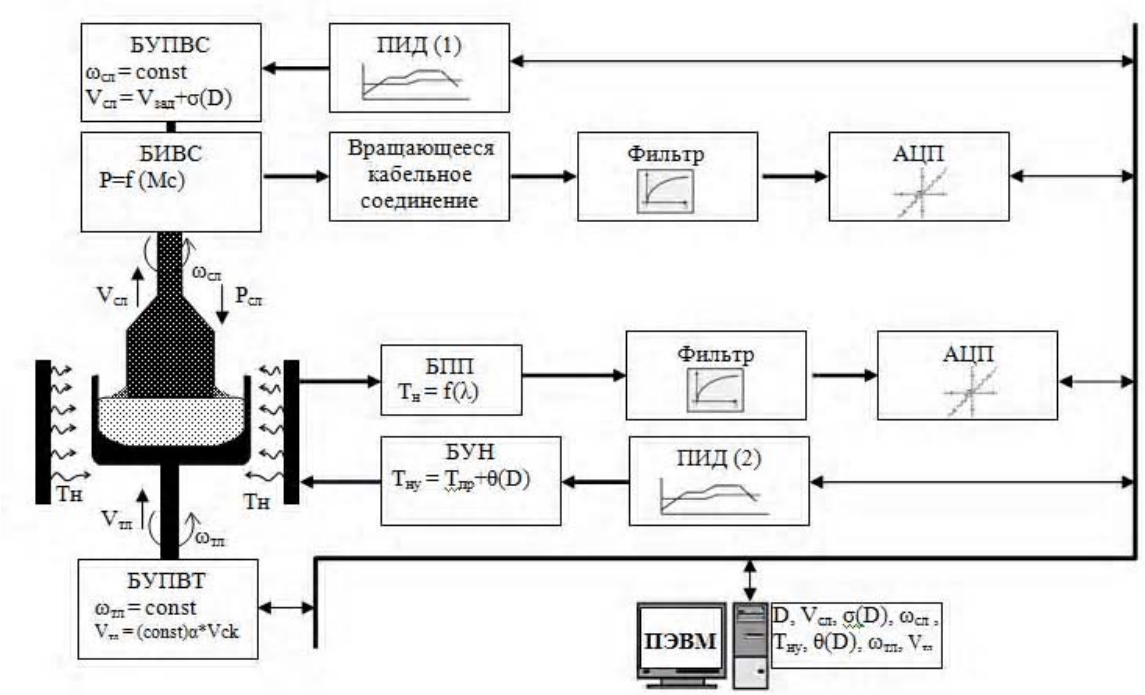


Рис. 2. Функциональная схема устройства автоматического контроля и поддержания диаметра слитка германия

БУПВТ – блок управления перемещением и вращением тигля. Управляет приводом вращения тигля и приводом скорости перемещения тигля.

БИВС – блок измерения веса слитка. Измеряет вес слитка, который определяется выражением : $P_{сл}(t) = m_0 g + \int_0^t \pi r^2 \rho_c g v_{сл} dt$, где m_0 — масса штока со слитком; r – радиус слитка; $v_{сл}$ – скорость подъема слитка.

БУПВС – блок управления перемещением и вращение слитка. Управляет приводом вращения слитка и приводом скорости перемещения слитка.

БУН – блок управления нагревателем. Осуществляет управление графитовым нагревателем.

БПП – блок пирометрического преобразователя. Измеряет температуру нагревателя и преобразует ее в напряжение.

ПИД – пропорционально-интегрально-дифференциальный регулятор. Осуществляет поддержку температуры или скорости перемещения слитка в заданном значении.

АЦП – аналого-цифровой преобразователь.

ПЭВМ – промышленная ЭВМ. Вычисляет диаметр слитка по данным БИВС.

Как было показано выше, управление диаметром слитка можно осуществлять регулируя скорость выращивания слитка $V_{сл}$ и мощность нагревателя $P_н$.

При отсутствии возбуждающих факторов для роста слитка с постоянным диаметром требуется постоянная скорость выращивания (формируемая блоком ПИД(1)) и программное изменение мощности нагревателя (формируемой блоком ПИД(2)), компенсирующее изменение градиента температуры вследствие изменения массы расплава.

В реальных условиях роста слитка и для скорости выращивания, и для мощности нагревателя требуется корректировка скорости выращивания в зависимости от отклонения диаметра от заданного ($\sigma(D)$) в соответствии с выражением (8), а также корректировка мощности нагревателя ($\theta(D)$) в соответствии с выражением (9).

Задача вычисления корректирующих воздействий решается ПЭВМ, которая вычисляет значение текущего диаметра слитка по выражению (12), формирует заданные значения

скорости выращивания слитка $V_{сл}$, вычисляет корректирующие значения $\sigma(D)$ и $\theta(D)$, вычисляет скорость подъема тигля, формирует программные значения для мощности нагревателя $T_{ну}$, частоты вращения тигля $\omega_{тл}$ и слитка $\omega_{сл}$.

5. Выводы

1. Разработана математическая модель и схема теплового узла с системой сервоконтроля с замкнутой петлей обратной связи, в основу которой положен метод взвешивания выращиваемого слитка германия.

2. На основе разработанной математической модели теплового узла разработана функциональная схема устройства автоматического контроля и поддержания диаметра выращиваемого слитка германия, с помощью которой осуществляется контроль и регулировка всех параметров, определенных математической моделью с учетом значения регулируемого диаметра.

3. Практическая ценность полученных результатов заключается в существенном сокращении потерь дорогостоящего монокристалла германия при механической калибровке слитков и повышении структурных характеристик выращенных монокристаллов.

Список литературы: 1. *Канлунов И.А., Колесников А.И., Долнатов А.Б., Ткач О.И.* Механические напряжения и оптические аномалии в кристаллах германия и параттеллурита // Весник ТвГУ. Серия «Физика», 2004. №4(6). С.72–80. **2.** *Bardsley W., D.T.J. Hurle, G.C. Joyce*, The weighing of automatic crystal growth, Journal of Crystal Growth 40 (1977). P. 13–20. **3.** *Tom H. Johansen*, The weight gain in Czochralski crystal growth, Journal of Crystal Growth 118 (1992). P. 353–359. **4.** *Winkler J., M. Neubert, J. Rudolph*, Nonlinear model-based control of the Czochralski process II: reconstruction of crystal radius and growth rate from the weighing signal, Journal of Crystal Growth 31 232 (2010). P. 1019–1028. **5.** *Rossolenko S., Pet'kov I., KurlovV., B. Red'kin*, Servo-controlled crystal growth by the Czochralski method estimating the state vector of the controlled object, Journal of Crystal Growth 116 (1992). P. 185–190. **6.** *Winkler J., Neubert M., Rudolph J.,* Nonlinear model-based control of the Czochralski process I: motivation, modeling, and feedback controller design, Journal of Crystal Growth 312 (2010). P. 1005–1018. **7.** *Winkler J., Neubert M., Rudolph J.* Nonlinear model-based control of the Czochralski process I: motivation, modeling, and feedback controller design, Journal of Crystal Growth 312 (2010). P. 1005–1018.

Поступила в редколлегию 25.05.2012

Оксанич Анатолий Петрович, д-р техн. наук, профессор, директр 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 материаллов. Адрес: Украина, 39600, Кременчуг, ул. Первомайская, 20, тел. (05366) 30157. Email: pritchinse@ukr.net.

Малёваный Владимир Викторович, аспирант кафедры информационно-управляющих систем Кременчугского национального университета имени Михаила Остроградского. Научные интересы: автоматизация процессов управления производством полупроводниковых материаллов. Адрес: Украина, 39600, Кременчуг, ул. Первомайская, 20, тел. (05366) 30157. Email: ius.krnu@gmail.com.

РЕФЕРАТИ

УДК 621.3.06

Великі шифри - випадкові підстановки. Порівняння диференціальних та лінійних властивостей шифрів, представлених на Український конкурс, та їх зменшених моделей / І.В. Лисицька, А.О. Настенко, К.Є. Лисицький // АСУ та прилади автоматики. 2012. Вип. 159. С. 4-10.

Показано матеріали по додатковому обгрунтуванню справедливості гіпотези про те, що великі шифри асимптотично є випадковими підстановками. Виконано порівняння диференціальних і лінійних властивостей шифрів, представлених на український конкурс і їхніх зменшених моделей. Розглянуто показники фіналістів конкурсу AES шифру Rijndael, Serpent, а також Threefish. Встановлено, що за диференціальними і лінійними показниками українські шифри Калина, Мухомор і Лабіринт перевершують визнаного світового лідера блокового симетричного шифрування.

Табл. 8. Бібліогр. :13 назв.

UDC 621. 3.06

Large ciphers - the random permutation. Comparison of differential and linear properties of the block symmetric ciphers before the Ukrainian competition / I.V. Lisitskaya, A.A. Nastenko, K.E. Lisitsky // Management Information System and Devices. 2012. N 159. P.4-10.

Submissions on additional substantiation of the validity of the hypothesis that large codes are asymptotically random substitutions. A comparison is made of differential and linear properties of ciphers submitted to the Ukrainian competition and reduced models. Considered as linear and differential properties finalists AES: cipher Rijndael, Serpent, and Threefish. It is established that the differential and linear characteristics of the Ukrainian ciphers Kalina, Muhomor and Labirint exceed recognized world leader block symmetric encryption.

Tab. 8. Ref.: 13 items.

УДК 658.512.011:681.326:519.713

Технології відновлення працездатності мультипроцесорних систем на кристалах / Мурад Алі Аббас, Багхдаді Аммар Авні Аббас, В.І. Хаханов, Є.І. Литвинова, Дахіри Фарід // АСУ та прилади автоматики. 2012. Вип. 159. С. 10-23.

Наведено огляд мультипроцесорних систем на кристалах (MPSoC) і технологій відновлення їх працездатності. Описано архітектури MPSoC, базову архітектуру самовідновлення, метод виявлення та виправлення помилок в програмованих логічних матрицях FPGA.

Іл. 12. Бібліогр.: 50 назв.

UDC 658.512.011:681.326:519.713

Technologies for repairing multiprocessor systems-on-chips / Murad Ali Abbas, Bahdadi Ammar Avni Abbas, V.I. Hahanov, E.I. Litvinova, Dahiri Farid // Management Information System and Devices. 2012. N 159. P.10-23.

An overview of multiprocessor systems-on-chips (MPSoC) and technologies for their repairing is presented. The MPSoC architectures, the basic architecture of self-repairing, and method for detecting and correcting errors in programmable logic arrays FPGA are described.

Fig. 12. Ref.: 50 items.

УДК 621.372.061

Характеристики різних стратегій проектування аналогових кіл в розширеному базисі / О.М. Земляк, Т.М. Маркіна // АСУ та прилади автоматики. 2012. Вип. 159. С. 23-30.

Сформульована узагальнена методологія другого рівня для проектування аналогових кіл, яка заснована на вживання і теорії оптимального управління. Подальше дослідження цього питання може бути сфокусовано на проблемі пошуку мінімальної за часом стратегії проектування шляхом оптимізації вектора, що управляє, в розширеному базисі.

Табл. 4. Іл. 4. Бібліогр.: 8 назв.

UDC 621.372.061

Characteristics of different strategies of designing of analog circuits in extended basis / A.M. Zemliak, T.M. Markina // Management Information System and Devices. 2012. N 159. P.23-30.

The generalized methodology of the second level is formulated for designing of analog circuits. This methodology is based by application of theory of optimum control. Further research can be focusing on the problem of search of optimal strategy of designing by means of optimization of structure of control vector in the extended basis.

Tab. 4. Fig. 4. Ref.: 8 items.

УДК 638.562:51.65.012

Удосконалені математичні моделі опису характеристик операції замовлення і обладнання поліграфічного підприємства / І.В. Левикін, К.В. Логвиненко // АСУ та прилади автоматики. 2012. Вип. 159. С. 30-33.

Розглянуто існуючі постановки та вирішення задачі планування виконання замовлень поліграфічного підприємства. Розроблено вдосконалені математичні моделі опису характеристик операції замовлення і обладнання поліграфічного підприємства, які враховують особливості виробництва друкованих видань і можуть бути використані для вирішення задач відділів планування виробництва та планово-диспетчерських служб поліграфічних підприємств різних типів.

Бібліогр.: 5 назв.

UDC 638.562:51.65.012

Advanced mathematical model describing characteristics of order's operations and equipment of printing companies / I.V. Levikyn, K.V. Logvynenko // Management Information System and Devices. 2012. N 159. P.30-33.

The existing formulation and solution of the scheduling printing copmany orders are considered. The sophisticated mathematical models, which describe the characteristics of the order's operation and equipment of printing companies, and take into account the peculiarities of the printing productiont and can be used for solving the tasks of planning department, planning and dispatching services in printing companies of various types are developed.

Ref.: 5 items.

УДК 519.7

Про підхід до побудови ланцюгів лексичних одиниць української мови в лексикографічній системі електронного тлумачного словника / Т.М. Федорова // АСУ та прилади автоматики. 2012. Вип. 159. С. 33-40.

Розглянуто подальший розвиток методу знаходження n-го лінійного логічного перетворення для побудови ланцюгів в лексикографічній системі електронних тлумачних словників. Модифікація методу характеризується завданням початкової семантичної залежності на кожному кроці. Також розглянута реалізація методу програмою «Побудова гіперланцюгів», яка дозволяє будувати, редагувати та аналізувати ланцюги.

Іл. 5. Бібліогр.: 5 назв.

UDC 519.7

About the approach to creation of chains lexical units of ukrainian in lexicographic system of the electronic explanatory dictionary / T.N. Fyodorova // Management Information System and Devices. 2012. N 159. P.33-40.

In article further development the method of finding n of linear logic transformation for creation of chains in lexicographic system electronic explanatory dictionaries is considered. Updating of a method is characterized by a task of initial semantic dependence at each stage of calculation. Also in article method realization by the program «Pobudova Giperlanzugiv» which allows to build, edit and analyze chains is considered.

Fig. 5. Ref.: 5 items.

УДК 65.011.56

Модель представлення варіантів технологічних процесів у базі прецедентів / В.О. Філатов, Р.В. Артюх // АСУ та прилади автоматики. 2012. Вип. 159. С. 40-45.

Запропоновано структурні моделі представлення технологічних процесів для формування архіву аналогів технологічних рішень. На основі технологічної операції визначається принцип уніфікованої деталі і групового технологічного процесу обробки, що дає можливість використати спосіб компактного, інформативного і наочного представлення структури пооперaційного технологічного процесу. Вдосконалена модель представлення варіантів технологічних процесів у базі прецедентів, що дозволяє зменшити витрати часу на пошук інформації для прийняття рішень.

Табл. 1. Іл. 3. Бібліогр.: 9 назв.

UDC 65.011.56

Model Of Presentation Of Variants Technological Processes In Base Of Precedents / Filatov V., Artyukh R. // Management Information System and Devices. 2012. N 159. P.40-45.

The structural models of presentation of technological processes are offered for forming of archive of analogues of technological decisions. On the basis of technological operation principle of compatible detail

and group technological process of treatment is determined, that gives an opportunity to use the method of compact, informing and evident presentation of structure of technological process. The model of presentation of variants of technological processes is improved in the base of precedents, that allows to bring down the expenses of time on an information retrieval for making decision.

Tab. 1. Fig. 3. Ref.: 9 items.

УДК 519-866

Моделювання фінансових ризиків з використанням ймовірнісного підходу / О.А. Кожухівська // АСУ та прилади автоматики. 2012. Вип. 159. С. 46-53.

Визначено типи сучасних актуарних ризиків, які потребують аналітичного дослідження за допомогою математичних і статистичних моделей різної структури. Встановлено методи кількісного аналізу для розв'язання задачі поглибленого розуміння суті та оцінювання рівня фінансового ризику. Для оцінювання ймовірності втрат успішно застосовано моделі імовірнісного типу, оскільки вони дають можливість враховувати параметричні і статистичні невизначеності досліджуваного процесу. Розглянуто байєсівський підхід до опису ризиків і запропоновано процедуру формування моделі імовірнісного типу, яка використана для побудови прогнозуючої моделі стосовно надходжень платежів до страхової компанії. Виконано обчислювальні експерименти з метою оцінювання параметрів прогнозуючого розподілу. Отриманий результат близький до результатів застосування методу моментів до фактичних статистичних даних.

Бібліогр.: 14 назв.

UDC 519-866

Modeling financial risks using probabilistic approach / O.A. Kozhukhivska // Management Information System and Devices. 2012. N 159. P.46-53.

The types of modern actuarial risks are determined that require of analytical study with application of mathematical and statistical models with various structures. There were established the types of quantitative methods for deeper understanding and estimation of financial risk level. For estimating the loss probability the probabilistic type models are used due to the possibility of taking into consideration parametric and statistical uncertainties of a procees under study. A Bayesian approach to modeling was considered and the procedure for constructing probabilistic type model is proposed. The procedure is used to model the data characterizing a stream of company payments. A computing experiment was performed aiming to estimate forecasting distribution parameters. The result achieved is close to the one produced by the method of moments.

Ref.: 14 items.

УДК 681.326:519.613

Метод підвищення контролепридатності критичних систем керування АЕС/ К.Є. Герасименко // АСУ та прилади автоматики. 2012. Вип. 159. С. 53-57.

Запропоновано метод підвищення контролепридатності обладнання захисту керуючої системи безпеки АЕС, який характеризується використанням функціональних елементів (порівняння з уставкою, «і», «або», «2 з 4»), побудованих на базі арифметичних операцій без використання логічних команд. Метод дозволяє контролювати працездатність елементів захистів за їх реакцією на зміни вхідного безперервного сигналу і орієнтований на контроль та діагностування прихованих несправностей типу «неспрацювання».

Табл. 1. Бібліогр.: 19 назв.

UDC 681.326:519.613

Method of NPP critical control systems diagnosability improving / K.E. Gerasimenko // Management Information System and Devices. 2012. N 159. P.53-57.

A method for improving diagnosability of NPP control safety system is proposed. It is characterized by using functional elements (compared to the set point, "and", "or", "2 of 4"), based on arithmetic operations without using logical instructions. The method allows managing the efficiency of security features in their response to changes in the input continuous signal and it is focused on detection and diagnosis of latent faults such as "failure on demand".

Tab. 1. Ref.: 19 items.

УДК 004.853

Експертна система семантичного пошуку релевантної інформації та формування адаптивних Web-сторінок / О.М. Почанський // АСУ та прилади автоматики. 2012. Вип. 159. С. 57-66.

У статті розглядається задача підвищення ефективності пошуку релевантних даних шляхом створення експертної системи, метою якої є отримання інформації, цікавої користувачу, і її відображення у вигляді документа, що складається з адаптивних Web-сторінок. Наведено результаті моделювання.

Табл. 1. Іл. 5. Бібліогр.: 3 назви.

UDC 004.853

Expert system for search of relevant information and synthesis of adaptive Web-pages / O.M. Pochanskiy // Management Information System and Devices. 2012. N 159. P.57-66.

This paper represents the decision of problem of relevant data search using the expert system. It's main goal is to search information, interesting to the user, and displaying it in the form of a document consisting of adaptive Web-pages.The results of modeling are presented.

Tab.1. Fig.5. Ref.: 3 items.

УДК 519.7:007.52

Удосконалення технологій доступу й обробки зв'язаних даних семантичних додатків LinkedData/ І.Н. Галушка, В.В. Завгородній, С.Н. Солошич, С.С. Щербак // АСУ та прилади автоматики. 2012. Вип. 159. С. 67-73.

Ріст популярності концепції зв'язаних даних поряд з описом інформації у вигляді триплетів RDF обумовив необхідність дослідження процедур взаємодії зі сховищами триплетів. У результаті розроблений універсальний пошуковий інтерфейс, що забезпечує візуальну побудову запитів до сховищ триплетів. Візуальні запити автоматично перетворяться в мову запитів SPARQL, що використовується для доступу до сховищ триплетів. Після виконання запиту користувач отримує контекст із триплетами, що відповідають шуканим з урахуванням заданих обмежень на предикати й об'єкти.

Іл. 4. Бібліогр.: 3 назви.

UDC 519.7:007.52

Perfection of processing technologies and access to linked data / I.N.Galushka, V. V. Zavgorodniy, S.N. Soloshish, S.S. Shcherbak // Management Information System and Devices. 2012. N 159. P.67-73.

The increase of linked data popularity along with information description in the form of RDF-triplets caused the necessity of studying the procedures for interaction with triple stores. A unique search interface have been developed, which enables visual querying the triple stores. These visual queries are automatically converted into SPARQL query language that is used for triple store accessing. After the query is executed, a user gets the desired context with triplets according to the set constraints for predicates and object.

Fig.4. Ref.: 3 items.

УДК 681.5+548.55

Розробка математичної моделі і пристрою автоматичного контролю та підтримки діаметра зливків германію, що вирощуються за методом Чохральського / А.П. Оксанич, С.Е. Притчин, В.В. Мальований// АСУ та прилади автоматики. 2012. Вип. 159. С. 73-79.

Запропонована математична модель автоматизованої системи контролю, а також розроблена функціональна схема пристрою вирощування злитків германію. Запропоновані рішеня дозволили скоротити втрати германію при виконанні операції калібрування.

Іл. 2. Бібліогр.: 7 назв.

UDC 681.5+548.55

Development of mathematical models and automatic control and maintain Diameter Ingots Germany grown by the Czochralski method / A.P. Oksanych, S.E. Pritchin ,V.V. Malovany // Management Information System and Devices. 2012. N 159. P. 73-79.

The paper presents a mathematical model of the automated control system and developed a functional block diagram of the growing germanium ingots. Proposed solutions have reduced the loss of germanium when performing calibration.

Fig. 2. Ref.: 7 items.

ПРАВИЛА

оформления рукописей для авторов научно-технического сборника *"АСУ и приборы автоматики"*

Формат страницы — А4 (210х297мм), поля: сверху, справа, слева, снизу – 30 мм. Редактор: Pagemaker 6.0, 6,5 (можно, но нежелательно Word), гарнитура Times New Roman Cyr, кегль – 11 пунктов, межстрочное расстояние — 110 %, табуляция — 5 мм.

Объем рукописи – до 10 с. (языки: русский, украинский, английский). Содержание должно отражать актуальность исследования, постановку задачи, цель, сущность, научные и практические результаты, сравнение с лучшими аналогами, выводы.

Структура рукописи: заголовок, аннотация, текст, литература, реферат на украинском и английском языках, сведения об авторах.

ОБРАЗЕЦ ОФОРМЛЕНИЯ

УДК 519.713

И.О. ФАМИЛИЯ

НАЗВАНИЕ РУКОПИСИ

Аннотация (абзац 5-10 строк, кегль 10) помещается в начале статьи и содержит информацию о результатах описанных исследований.

Основной текст можно разделять на 2 и более подразделов с заголовками, выделенными полужирным шрифтом, пронумерованными арабскими цифрами, как показано в следующей строке.

1. Название раздела

Рисунки и таблицы (черно-белые, контрастные) помещаются в текст после первой ссылки в виде *переносимых объектов* и раздельно нумеруются, при наличии более одного рисунка (таблицы), арабскими цифрами. Рисунок содержит подрисуночную центрированную подпись (текстовая строка, расположенная вне рисунка, кегль 10) под иллюстрацией, как показано на рис. 1.

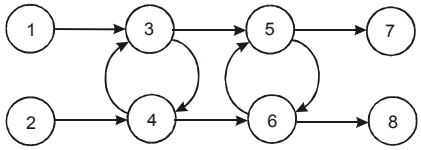


Рис. 1. Граф с контурами

Табличный заголовок располагается справа над таблицей, что иллюстрируется табл.1. Редакторы: CorelDraw, Table Editor и др.

Таблица 1

Шаг i	1	2	3	4	5	6
Ф1(1,3)	1	2	2	4	6	1

Формулы нумеруются при наличии ссылок на них в рукописи. Рекомендуемый кегль формульного набора: обычный (переменная) – 11 пунктов, крупный индекс – 8, мелкий индекс (над- и подиндекс) – 8, крупный символ (основной) – 12, мелкий (индексный) математический символ – 10:

$$F_{i+j} = \sum_{i=1}^{b^k} F_j^i - \prod_{j=1}^{1+h^2} P_{R\,j+i} + F^{j-1} + X^{\Sigma n^k} . \qquad (1)$$

Формат переменных (желательно не курсивом – без наклона) в тексте и формулах должен быть идентичным. В тексте над- и подиндексы составляют 70 % от кегля, которые рекомендуется опускать (поднимать) на 17 (33) % относительно основной строки.

Список литературы (включает опубликованные источники, на которые имеются ссылки в тексте, заключенные в квадратные скобки) печатается без отступа, кегль 9 пунктов.

Образец окончания текста рукописи (литература, сведения об авторах, реферат) представлен ниже.

Список литературы: 1. *Фамилия И.О.* Название книги. Город: Издательство, 1900. 000 с. 2. *Название* сборника / Под ред. И.О. Фамилия. Город: Издательство, 1900. 000 с. 3. *Фамилия И.О.* Название статьи / / Название журнала. Название серии. 1997. Т. 00, № 00. С. 00-00 .

Поступила в редколлегию 00.00.00

Фамилия, имя, отчество, ученая степень, звание, должность и место работы. Научные интересы. Адрес, контактный телефон.

Рефераты на украинском и английском языках:

УДК 000.000.00

Назва статті українською мовою / Ініціали. Прізвище // АСУ та прилади автоматики. 2000. Вип. 00. С. 000-000.

Текст реферату.

Табл. 00. Іл. 00. Бібліогр.: 00 назв.

UDC 000.000.00

Title of paper / Initials. Surname // Management Information System and Devices. All-Ukr. Sci. Interdep. Mag. 2000. N 00. P. 000-000.

Text.

Tab. 00. Fig. 00. Ref.: 00 items.

Представление материалов

Рукопись, реферат, сведения об авторах — в одном файле, *поименованном фамилией первого автора*, на дискете 3,5 дюйма. Твердая копия материалов – для граждан Украины — в одном экземпляре: рукопись, подписанная авторами, рефераты, акт экспертизы, внешняя рецензия, подписанная доктором наук, заявление на имя главного редактора со сведениями об авторах.

Адрес редакции: Украина, 61166, Харьков, пр. Ленина, 14, ХНУРЭ, комната 321, тел. 70-21-326, e-mails: ri@kture.kharkov.ua; hahanov@kture.kharkov.ua. http://www.ewdtest.com/ri

Тематика статей, публикуемых в сборнике:

– Компьютерная инженерия
– Математическое моделирование
– Оптимизация и процессы управления
– Автоматизация проектирования и диагностика
– Информационные интеллектуальные системы
– Проектирование интегральных схем и микросистем
– Компьютерные технологии в образовании

Відповідальний випусковий В.І. Хаханов
Редактор О.П. Гужва
Комп'ютерна верстка Г.В. Хаханова, С.В. Чумаченко

Підп. до друку 27.06.2012. Формат 60x841/8. Умов. друк. арк. .
Обл.-вид. арк. 10,1. Тираж 300 прим.
Зам. № б/н. Ціна договірна.

Харківський національний університет радіоелектроніки (ХНУРЕ).
Україна, 61166, Харків, просп. Леніна, 14.

Оригінал-макет підготовлено в навчально-науковому видавничо-поліграфічному центрі ХНУРЕ
Україна, 61166, Харків, просп. Леніна, 14.
Надруковано у видавництві ПП "Степанов В.В."
61168, Харків, вул. Акад. Павлова, 311