

ливо $g \approx m^2/2$ і $m \approx \sqrt{2g}$. Підставляючи останній результат m в вираження для числа точок, отримуємо

$$N_g(q) = 2\sqrt{2g} + 4(1 + \langle (q-1)/(4g^2) - 2 \rangle_3)g + 6g.$$

Границя Хассе – Вейля для максимальних кривих равна $N_g(q)_{HV} = q + 1 + 2g\sqrt{q}$. Вихислення $\limsup_{g \rightarrow \infty} N_g(q)/N_g(q)_{HV}$ дає (6). \diamond

Точні вихислення $N_g(q)/g$ згодуються з результатами (5) і (6). Так, для $q = 257$ і $m = 16$ маємо $N = 816$ і $N_g(q)/g = 7,76$, а для $q = 2^{16} + 1 = 65537$ і $m = 2048$ маємо $N = 12589056$ і $N_g(q)/g = 6,01$.

ВИВОДИ

1. В простому полі не існує максимальних кривих Ферма. При великому роді проигрыш границі Хассе – Вейля пропорційний $1/(\sqrt{q})$. С зменшенням роду кривої значення числа точок наближається до границі Хассе – Вейля, і при $g = 0,1$ маємо тривіальний випадок $N = q + 1$.

2. Універсальне хешування по кривим Ферма в простому полі не забезпечує вимоги по ймовірності колізії. В відповідності з вираженням (2) для ϵ алгебраїчна крива повинна бути великого роду, з великим числом точок і значущим відношенням $N_g(q)/g$. Асимптотика $N_g(q)/g$ для кривих Ферма равна 10. Криві малого роду проигривають по параметру N , що не забезпечує вимоги по ймовірності колізії для практичних значень q .

3. Результати для кривих Ферма представлені для простого поля і частково можуть бути віднесені

до властивостей кривих в розширеннях кінцевого поля. Розширені кінцеві поля мають більше багатозначності по комбінаторним властивостям, що впливає на оцінку параметрів кривих.

СПИСОК ЛІТЕРАТУРИ

1. Bierbrauer, J. On families of hash functions via geometric codes and concatenation. / Bierbrauer J., Johansson T., Kabatianskii G., Smeets B. // Advances in Cryptology-CRYPTO'93 Proceedings. – Springer-Verlag, 1994. – P. 331–342.
2. Халимов, Г. З. Колізіонні оцінки універсального хешування на основі схем з алгебраїчними кодами / Халимов Г. З. // Прикладна радіоелектроніка. – 2009. – Т. 8, Вип. 3. – С. 338–342.

Надійшла 25.10.2010

Халимов Г. З.

ОЦІНКА ПАРАМЕТРІВ КРИВИХ ФЕРМА ДЛЯ УНІВЕРСАЛЬНОГО ГЕШУВАННЯ

Отримано точні рішення для числа точок кривих Ферма, коли порядок поля має дільники 2, 3 та 6, також оцінки числа точок на основі ймовірнісного підходу. Приводяться асимптотичні границі відношення максимального числа точок кривої Ферма в простому полі до її роду та до границі Хассе – Вейля.

Ключові слова: універсальне гешування, криві Ферма.

Khalimov G. Z.

ESTIMATION OF FERMA CURVES PARAMETERS FOR UNIVERSAL HASHING OF NUMBER SOLUTION FOR HURVITZ EQUATION IN THE FINITE FIELD

Exact solutions for a number of Fermat curves points when the field order has divisors 2, 3 and 6 as well as estimates of points number have been obtained using probabilistic approach. Asymptotic boundaries of the ratio of maximum number of Fermat curve points in a simple field to its genus and to Hasse–Weil boundary are given.

Key words: universal hashing, Fermat curves.

УДК 004

Овсяк О. В.

Канд. техн. наук, доцент Львівської філії Київського національного університету культури і мистецтв

МОДЕЛЬ РОЗШИРЕНОЇ НОТАЦІЇ ТЕКСТОВОГО ОПИСУ ФОРМУЛ АЛГОРИТМІВ

З метою запису у пам'яті комп'ютера графічно-текстових формул алгоритмів, які утворені функційними унітермами з впорядкованими змінними і параметрами, створено розширену **xml**-подібну нотацію опису формул алгоритмів. Модель розширеної нотації описана засобами алгебри алгоритмів з використанням операції секвенування. Наведено приклад використання розширеної нотації для опису формули алгоритму Евкліда.

Ключові слова: модель, нотація, синтаксис, семантика, унітерм, секвенція, елімінавання, паралелення.

ВСТУП

Для опису алгоритмів інформаційних технологій і систем найчастіше використовуються вербальний і

блок-схемний методи. Відомо, що ці методи, як і методи машин Тюрінга [1], Поста [2], Колмогорова [3], Ахо – Ульмана – Хопкрофта [4], Шонхаге [5] і рекур-

© Овсяк О. В., 2010

сивних функцій [6], алгоритмів Маркова [7] та універсальних алгоритмів Крініцького [8] є методами неформального опису алгоритмів [9]. З метою математичного опису алгоритмів інформаційних технологій і систем створена алгебра алгоритмів [10] та її розширення [11]. Операції алгебри алгоритмів є оригінальними і мають специфічні позначення, яких немає серед стандартних математичних позначень. Тому для автоматизації процесів набору і редагування формул алгоритмів потрібно створити спеціалізовану комп'ютерну систему.

Як правило, кожна із комп'ютерних систем має свій формат даних, наприклад, **Word** використовує формат даних з розширенням **doc**. Останнім часом одним з найбільш часто використовуваних форматів є **xml** формат. На основі мови опису даних **XML** [12] створена сучасна мова **XAML** [13], призначена для програмування інтерфейсів прикладних інформаційних і комп'ютерних систем та комп'ютерної графіки. Нотація (система умовних письмових позначень [14]) **XAML**, у порівнянні з сучасною мовою об'єктного програмування **C#** [12], у більшій мірі зорієнтована на розробників комп'ютерної графіки.

ФОРМУЛЮВАННЯ ЗАДАЧІ

Спеціалізованими редакторами для набору і редагування формул алгоритмів є комп'ютерні системи **МОДАЛ** [15], **АБСТРАКТАЛ** [16] і **GenCod** [17]. Системи **МОДАЛ** і **АБСТРАКТАЛ** хоч і відрізняються своїми можливостями, але мають подібні формати даних. Подібність форматів даних полягає в наявності полів, які призначені для зберігання координат розташування графічних знаків операцій на робочому полі системи. Недоліком такого розв'язання є жорстка фіксація формул алгоритмів до місця розташування на робочому полі комп'ютерної системи. Для усунення цього недоліку у системі **GenCod** створено **xml**- подібний формат даних збереження формул алгоритмів у пам'яті комп'ютера. Однак формат **GenCod** має такий недолік, як відсутність можливості впорядкування змінних функційних унітермів. З метою усунення цього недоліку у статті введено модифікований формат текстового опису формул алгоритмів.

МОДЕЛЬ РОЗШИРЕНОЇ НОТАЦІЇ ОПИСУ ФОРМУЛ АЛГОРИТМІВ

Опис моделі розширеної нотації виконаємо засобами алгебри алгоритмів. Загальний вигляд текстового опису формули алгоритму є таким:

$$L; \overbrace{R_1; Q; R_2}$$

де

$$L = \overbrace{\langle ?xml; \#; v; \#; z; \#; ? \rangle}$$

де $\langle ?xml$ – константа, ідентифікуюча початок опису заголовку формули алгоритму, $\#$ – наявність одного або декількох пропусків (пробілів), v – версія формату, z – формат кодування, $?>$ – константа, ідентифікуюча кінець опису заголовку формули алгоритму,

$$v \in B = \overbrace{version="1.0", version="1.1", \dots, version="X.Y"}$$

$$z \in Z = \overbrace{encoding="utf8", encoding="utf16"}$$

$$R_1 = \overbrace{\langle ; root; \rangle}$$

\langle – константа-ідентифікатор початку опису дескриптора, $root$ – ідентифікатор кореневого дескриптора, \rangle – ідентифікатор закриття початку і кінця опису дескриптора,

$$R_2 = \overbrace{\langle /; root; \rangle}$$

де $\langle /$ – константа-ідентифікатор опису кінця дескриптора,

$$Q = \overbrace{W_0; W_1; \dots; W_{n-1}}, Q = \overbrace{W_0; W_1; \dots; W_{n-1}}$$

$$W_i = \overbrace{V_0; V_1; \dots; V_{m-1}}, i \in \overbrace{0; 1; \dots; n-1}$$

$$V_j \in M = \overbrace{S, E, P, C^s, C^e, C^p, U, *}$$

$$j \in \overbrace{0; 1; \dots; m-1}$$

де $S, E, P; C^s, C^e, C^p, U$ – граматики опису операцій секвентування, елімінування, паралелення, циклічного секвентування, циклічного елімінування, циклічного паралелення і унітерма, відповідно, a^* – порожній унітерм.

Синтаксис і семантика опису унітермів

Унітерми є одного із трьох типів:

$$U \in \overbrace{U_1, U_2, U_3}$$

де U_1 – тривіальний (несеквенційний) унітерм, U_2 – секвенційний унітерм, U_3 – подвійний унітерм.

1. Опис тривіальних (несеквенційних) унітермів має такий вигляд:

$$U_1 = \overbrace{\langle ; uniterm; \#; u; \#; / \rangle}$$

де $uniterm$ – ідентифікатор опису унітерма, u – унітерм (будь-які символи та їх послідовності).

2. Такою формулою описуються секвенційні унітерми:

$$U_2 = \langle ; \overbrace{\text{uniterms}} ; \# ; \overbrace{u_1} ; \# ; \overbrace{u_2} ; \# ; / \rangle ,$$

де *uniterms* – ідентифікатор опису секвенційного унітерма, *u_1* – знак (позначення секвенційного унітерма) і *u_2* – секвенційні змінні унітерма.

3. Формат подвійних унітермів:

$$U_3 = \langle ; \overbrace{\text{unitermp}} ; \# ; \overbrace{u_1} ; \# ; \overbrace{u_2} ; \# ; \overbrace{u_3} ; \# ; / \rangle ,$$

де *unitermp* – ідентифікатор подвійного унітерма, *u_1* – знак подвійного унітерма, *u_2* – знак секвенційного унітерма і *u_3* – секвенційні змінні унітерма.

Опис складових операцій

А) Секвентування починається і закінчується такими секвенціями:

$$S_1 = \langle ; \overbrace{s_0} ; \# ; \overbrace{r} ; \# ; \overbrace{o} ; \rangle ,$$

$$S_2 = \langle / ; \overbrace{s_0} ; \rangle ,$$

де *s_0* = "sequence" – ідентифікатор опису секвентування, *r* – розділювач унітермів, *o* – орієнтація знаку операції секвентування,

$$z \in R = \overbrace{\text{"semicolon"}, \text{"coma"}},$$

$$o \in O = \overbrace{\text{"horizontal"}, \text{"vertical"}},$$

де "semicolon" – розділювач крапка з комою, "coma" – розділювач кома, "horizontal" – орієнтація горизонтальна, "vertical" – орієнтація вертикальна.

Б) Елімінування має такий початок і кінець опису:

$$E_1 = \langle ; \overbrace{e_0} ; \# ; \overbrace{o} ; \rangle ,$$

$$E_2 = \langle / ; \overbrace{e_0} ; \rangle ,$$

де *e_0* = "elimination" – ідентифікатор опису елімінування.

В) Паралелення починається і закінчується такими секвенціями:

$$P_1 = \langle ; \overbrace{p_0} ; \# ; \overbrace{r} ; \# ; \overbrace{o} ; \rangle ,$$

$$P_2 = \langle / ; \overbrace{p_0} ; \rangle ,$$

де *p_0* = "parallelization" – ідентифікатор опису паралелення.

Г) Циклічне секвентування має такі секвенції:

$$C^s_1 = \langle ; \overbrace{c^s} ; \# ; \overbrace{o} ; \rangle ,$$

$$C^s_2 = \langle / ; \overbrace{c^s} ; \rangle ,$$

де *c^s* = "cyclic-sequence" – ідентифікатор опису циклічного секвентування.

Г) Циклічне елімінування починається і закінчується такими секвенціями:

$$C^e_1 = \langle ; \overbrace{c^e} ; \# ; \overbrace{o} ; \rangle ,$$

$$C^e_2 = \langle / ; \overbrace{c^e} ; \rangle ,$$

де *c^e* = "cyclic-elimination" – ідентифікатор опису циклічного елімінування.

Д) У циклічне паралелення входять такі секвенції:

$$C^p_1 = \langle ; \overbrace{c^p} ; \# ; \overbrace{o} ; \rangle ,$$

$$C^p_2 = \langle / ; \overbrace{c^p} ; \rangle ,$$

де *c^p* = "cyclic-parallelization" – ідентифікатор опису циклічного паралелення.

Синтаксис і семантика операцій

А) Секвентування:

$$S = \overbrace{S_1 ; W_i ; W_j ; S_2} .$$

Б) Елімінування:

$$E = \overbrace{E_1 ; W_i ; W_j ; W_k ; E_2} .$$

В) Паралелення:

$$P = \overbrace{P_1 ; W_i ; W_j ; P_2} .$$

Г) Циклічного секвентування:

$$C^s = \overbrace{C^s_1 ; \# ; W_g ; \# ; W_i ; \# ; C^s_2} .$$

Г) Циклічного елімінування:

$$C^e = C^e_1; \#; W_g; \#; W_i; \#; C^e_2.$$

Д) Циклічного паралелення:

$$C^p = C^p_1; \#; W_g; \#; W_i; \#; C^p_2,$$

де

$$W_g \in u, \quad W_j = V_0; V_1; \dots; V_{k-1}, \\ j \in \overline{0; T; \dots; k-1}.$$

ПРИКЛАД ОПИСУ АЛГОРИТМУ ЕВКЛІДА

З використанням розширеної нотації формула алгоритму Евкліда [2] для обчислення двох натуральних чисел матиме такий вигляд:

$$K_1; K_2; K_3: \left(\begin{array}{l} \leftarrow x; y; \notin x; N; \notin y; N; \leq x; y - ? \\ \varnothing \\ = x; y; \rightarrow y; \backslash x \% r; r =; 0 \neq - ? \\ \vdots \\ \leftarrow x; x \\ \vdots \\ \leftarrow r; 0 \neq \end{array} \right),$$

де K_1 – повідомлення про те, що змінна x не є натуральним числом, N – множина натуральних чисел; K_2 – повідомлення «Значення y не є натуральним числом», K_3 – повідомлення $x \leq y$, $\leftarrow x; y$ – унітерм введення з клавіатури значень змінних x і y ; $\notin x; N$ – перевірка неналежності значення змінної x до натуральних чисел; $\notin y; N$ – перевірка належності значення змінної y до натуральних чисел, $= x; y$, приписування змінній x значення змінної y , $x \% r$ – знаходження остачі від ділення x на y ;

$\leftarrow y; x \% r; r =; 0 \neq - ?$ – приписування змінній остачі від ділення і її порівняння з нулем; $\leftarrow r; 0 \neq$ – повернення у цикл за умовою $r \neq 0$, знак операції початку циклічного секвентування.

З використанням створеної моделі граматики текстового опису ця формула опишеться так:

```
<? xml version="1.0" encoding="utf-8"?>
<root>
<elimination orientation="horizontal">
<uniterm> K1; K2; K3</uniterm>
<sequence separator="semicolon" orientation="vertical">
```

```
<uniterms></uniterms>
<sequence separator="semicolon" orientation="horizontal">
<uniterm> x </uniterm>
<uniterm> y </uniterm>
</sequence>
<cyclic-sequence orientation="vertical">
<uniterm></uniterm>
<elimination orientation="horizontal">
<sequence separator="semicolon" orientation="vertical">
<uniterms>=</uniterms>
<sequence separator="semicolon" orientation="horizontal">
<uniterm> x </uniterm>
<uniterm> y </uniterm>
</sequence>
<sequence separator="semicolon" orientation="vertical">
<uniterms>=</uniterms>
<sequence separator="semicolon" orientation="horizontal">
<uniterm> y </uniterm>
<uniterm> r </uniterm>
</sequence>
<uniterms>c</uniterms>
<sequence separator="semicolon" orientation="horizontal">
<uniterm> r </uniterm>
<uniterm> 0 </uniterm> </uniterm>
</sequence>
<uniterms></uniterms>
</sequence>
</sequence>
<uniterms></uniterms>
<uniterms>y</uniterms>
<sequence separator="semicolon" orientation="horizontal">
<sequence separator="semicolon" orientation="horizontal">
<uniterm> y </uniterm>
<uniterm> x </uniterm>
</sequence>
<uniterms>%</uniterms>
<uniterms>r</uniterms>
</sequence>
<uniterms>=</uniterms>
<uniterm> 0 </uniterm>
</sequence>
<uniterms></uniterms>
</elimination>
</cyclic-sequence>
</sequence>
<sequence separator="semicolon" orientation="horizontal">
<uniterms>ь</uniterms>
<sequence separator="semicolon" orientation="horizontal">
<uniterm>x</uniterm>
<uniterm>N</uniterm>
</sequence>
<sequence separator="semicolon" orientation="horizontal">
<uniterms>ь</uniterms>
<sequence separator="semicolon" orientation="horizontal">
<uniterm>y</uniterm>
<uniterm>N</uniterm>
</sequence>
<uniterms>J</uniterms>
<sequence separator="semicolon" orientation="horizontal">
<uniterm>x</uniterm>
<uniterm>y</uniterm>
```

```

</sequence>
</sequence>
</sequence>
</elimination>
</root>

```

Версія і формат кодування описані у першому зверху рядку представленого *xml*-подібного формату. Другий рядок має назву кореневого дескриптора. Опис елімінування починається дескриптором третього рядка. У четвертому рядку описані унітерми K_1 , K_2 і K_3 . Опис дескриптора секвенчування починається у п'ятому рядку. Шостий рядок містить секвенційний унітерм введення значень змінних з клавіатури (\leftarrow). Самі ж назви змінних (x , y), яким приписуються введені з клавіатури значення, описані у сьомому, восьмому, дев'ятому і десятому рядках. Подальші рядки поданого прикладу досить легко читаються і тому не потребують додаткових пояснень.

ВИСНОВКИ

1. Створена модель розширеної нотації забезпечує текстовий опис формул алгоритмів як із тривіальними, так і секвенційними та дубльованими унітермами.

2. У моделі розширеної нотації міститься опис впорядкувань не тільки самих унітермів, а також і змінних унітермів.

3. Модель розширеної нотації забезпечує переносимість формул алгоритмів.

СПИСОК ЛІТРАТУРИ

1. *Turing, A. M.* On computable numbers, with an application to the Entscheidungsproblem / Turing A. M. // Proceedings of London Mathematical Society. – 1936–1937. – Series 2, Vol. 42. – pp. 230–265; correction, ibidem, vol. 43, pp. 544–546. – [Reprinted in [13 Davis M., pp. 155–222] and available online at <http://www.abelard.org/turpap2/tp2-ie.asp>].
2. *Post, E. L.* Finite Combinatory Processes - Formulation 1 / Post E. L. // Journal of Symbolic Logic. – 1936. – 1. – Pp. 103–105. – Reprinted in The Undecidable, pp. 289ff.
3. *Kolmogorov, A. N.* On the concept of algorithm (in Russian) / Kolmogorov A. N. // Uspekhi Mat. Nauk. – 1953. – 8:4. – Pp. 175–176. – [Translated into English in Uspensky V. A., Semenov A. L. Algorithms: Main Ideas and Applications. – Kluwer, 1993.]
4. *Aho, A. V.* The design and analysis of computer algorithms / Aho A. V, Hopcroft J. E, Ullman J. D. – Addison-Wesley Publishing Company, 1974.
5. *Schönhage, A.* Universelle Turing Speicherung / Schönhage A. // In J. Dörr and G. Hotz, Editors, Automatentheorie und Formale Sprachen. – Bibliogr. Institut, Mannheim, 1970. – Pp. 369–383.
6. *Church, A.* An unsolvable problem of elementary number theory / Church A. // American Journal of Mathematics. – 1936. – Vol. 58. – Pp. 345–363.
7. *Markov, A. A.* Theory of algorithms (in Russian) / Markov A. A. // Editions of Academy of Sciences of the USSR. – 1951. – Vol. 38. – Pp. 176–189 – [Translated in-

to English in American Mathematical Society Translations, 1960, series 2, 15, pp. 1–14.]

8. *Krinitiski, N. A.* Algorithms around us (in Russian) / Krinitiski N. A. – Moscow : Mir, 1988. – [Also translated to Spanish (Algoritmos a nuestro alrededor)].
9. *Успенский, В. А.* Теория алгоритмов: основные открытия и приложения / Успенский В. А., Семенов А. Л. – М. : Наука, 1987. – 288 с.
10. *Овсяк, В.* Засоби еквівалентних перетворень алгоритмів / Овсяк В. // Доповіді національної академії наук України. – 1996. – № 9. – С. 83–89.
11. *Owsiak, W.* Rozszerzenie algebry algorytmów / Owsiak W., Owsiak A. // Pomiar, automatyka, kontrola. – № 2. – S. 184–188.
12. *Дейтел, Х. С#* / Дейтел Х. и др. – М. ; СПб : БХВ-Петербург, 2006. – 1056 с.
13. *Мак-Дональд, М.* WPF. Windows Presentation Foundation in NET 3.5 с примерами на 2008 / Мак-Дональд М. – Второе издание. – М. : Вильямс, 2008. – 928 с.
14. Великий тлумачний словник сучасної української мови / уклад. і голов. ред. В. Т. Бусел. – К. ; Ірпінь: Перун, 2002. – 1440 с.
15. *Бритковський, В. М.* Моделювання редактора формул секвенційних алгоритмів: автореф. дис. на здобуття наук. ступеня канд. тех. наук: спец. 01.05.02 «Математичне моделювання та обчислювальні методи» / Бритковський В. М. – Львів, 2003. – 18 с.
16. *Василюк, А. С.* Підвищення ефективності математичного і програмного забезпечення редактора формул алгоритмів: автореф. дис. на здобуття наук. ступеня канд. тех. наук: спец. 01.05.02 «Математичне та програмне забезпечення обчислювальних машин і систем» / Василюк А. С. – Львів, 2008. – 20 с.
17. *Овсяк, О. В.* Класи інформаційної системи генерування коду / Овсяк О. В. // Вісник Тернопільського державного технічного університету. – 2010. – № 1. – С. 171–176.

Надійшла 06.07.2010

Овсяк А. В.

МОДЕЛЬ РАСШИРЕННОЙ НОТАЦИИ ТЕКСТОВОГО ОПИСАНИЯ ФОРМУЛ АЛГОРИТМОВ

С целью записи в память компьютера графически-текстовых формул алгоритмов, которые образованы функциональными унитермами с упорядоченными переменными и параметрами, создано расширенную *xml*-подобную нотацию описания формул алгоритмов. Модель расширенной нотации описана средствами алгебры алгоритмов с использованием операции секвенцирования.

Приведен пример использования расширенной нотации для описания формулы алгоритма Евклида.

Ключевые слова: модель, нотация, синтаксис, семантика, унитерм, секвенцирование, элиминирование, параллелирование.

Ovsyak O. V.

MODEL OF EXTENDED NOTATION OF ALGORITHM FORMULAS TEXTUAL DESCRIPTION

The proposed extended *xml*-notation of algorithm formulas description provides writing of graphical-textual algorithm formulas formed by functional uniterms with ordered variables and parameters in the computer memory. The extended notation model is described by means of algorithm algebra using the sequencing operation.

The example of Euclid algorithm formula description using the extended notation is given in the paper.

Key words: model, notation, syntax, semantics, uniterm, sequencing, elimination, paralleling.

НЕЙРОІНФОРМАТИКА ТА ІНТЕЛЕКТУАЛЬНІ СИСТЕМИ

НЕЙРОІНФОРМАТИКА И ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ

NEUROINFORMATICS AND INTELLIGENT SYSTEMS

УДК 004.021:004.622:65.021.1

Дубровин В. И.¹, Твердохлеб Ю. В.²

¹Канд. техн. наук, профессор Запорожского национального технического университета

²Студент Запорожского национального технического университета

УСОВЕРШЕНСТВОВАНИЕ МЕТОДОВ АНАЛИЗА ЭКГ-СИГНАЛОВ НА ОСНОВЕ ВЕЙВЛЕТ-ПРЕОБРАЗОВАНИЯ В СИСТЕМЕ ЭЛЕКТРОКАРДИОГРАФИИ ВЫСОКОГО РАЗРЕШЕНИЯ

Предложен метод анализа ЭКГ-сигналов на основе вейвлет-преобразования в системе электрокардиографии высокого разрешения. Данный метод позволяет обнаружить девять важных координат точек кардиосигнала: начало, пик и смещение Р-зубца, QRS-комплекса и Т-зубца во всех отведениях ЭКГ-сигнала. Предлагаемый подход увеличивает точность обнаружения Р и Т зубцов. Обоснован оптимальный способ фильтрации ЭКГ-сигнала от шумов.

Ключевые слова: электрокардиограмма, Р-QRS-Т-комплексы, вейвлет-преобразование.

ВВЕДЕНИЕ

Электрокардиограмма (ЭКГ) представляет собой запись электрических потенциалов активности сердца, снятую с одного или нескольких отведений, и состоит из периодической последовательности кардиоциклов. В типичном кардиоцикле выделяют несколько элементов: Р-волна, QRS-комплекс и Т-волна (рис. 1). Отправной точкой ряда современных методик компьютерной электрокардиографии является выделение положения QRS-комплекса, который отражает процесс деполяризации желудочков. Затем обнаруживают Р и Т зубцы, отражающие процесс деполяризации правого и левого предсердий, и процесс конечной реполяризации миокарда желудочков соответственно [1–3].

В настоящее время для анализа ЭКГ-сигнала перспективно применение вейвлет-анализа. Вейвлеты – это обобщенное название семейств математических функций определенной формы, которые локальны по

времени и по частоте, и в которых все функции получают из одной базовой (порождающей) функции посредством ее сдвигов и растяжений по оси времени. По сравнению с разложением сигналов на ряды Фурье, вейвлеты способны с гораздо более высокой точностью представить локальные особенности сигналов, вплоть до разрывов первого рода [4–5].

Целью данной работы является усовершенствование методов анализа ЭКГ-сигналов в системе электрокардиографии высокого разрешения.



Рис. 1. Типичный комплекс ЭКГ-сигнала

1. ВЫБОР ВИДА ВЕЙВЛЕТ-ПРЕОБРАЗОВАНИЯ И ТИПА ВЕЙВЛЕТ-НОСИТЕЛЯ

Разработанный метод анализа ЭКГ-сигнала базируется на непрерывном вейвлет-преобразовании (CWT). Непрерывное вейвлет-преобразование в разных временных масштабах характеризует сигнал в различных частотных диапазонах, в то время как дискретное вейвлет-преобразование (DWT) ограничено масштабами, которые являются степенями двойки. Использование CWT вместо DWT предоставляет больше возможностей [4–5].

Пусть s – сигнал и ψ – вейвлет. При непрерывном преобразовании вейвлет-коэффициенты сигнала s , соответствующие масштабному коэффициенту a и положению b , определяются формулой (1) [4]:

$$CWT(a, b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} s(t) \psi^*\left(\frac{t-b}{a}\right) dt. \quad (1)$$

Для выбора оптимального вейвлета, используемого в качестве базиса, было протестировано несколько вейвлет-функций. Под оптимальным вейвлетом будем иметь в виду тот, который обеспечивает правильное местоположение координат девяти точек кардиоцикла: начало, пик и смещение Т-зубца, QRS-комплекса и Р-зубца. В работах [6, 7] применяются биортогональные вейвлеты с компактным носителем, используя масштабы, кратные степени двойки. В работах [7, 8] используются гауссовы вейвлеты (пред-

вейвлеты). Свойства этих семейств вейвлетов [4, 5] представлены в табл. 1.

Наилучшим вейвлет-носителем, удовлетворяющим перечисленным выше требованиям, согласно полученным результатам (табл. 2), является биортогональный вейвлет «bior1.5».

В качестве масштаба, используемого для определения правильного местоположения девяти координат точек ЭКГ-сигнала, был использован 15 масштаб для обнаружения QRS-комплекса [7–11] и 41 масштаб для обнаружения Р и Т зубцов [7]. Масштабы 15 и 41 обеспечивают наибольшую точность в обнаружении данных зубцов. Вейвлет «bior1.5» в масштабах 15 и 41 представлен на рис. 2.

2. МЕТОД АНАЛИЗА КАРДИОСИГНАЛА

Процесс анализа кардиосигнала условно можно разделить на две стадии: стадия предварительной обработки и выделение признаков (рис. 3).

Стадия предварительной обработки заключается в удалении шумов (электромиографические потенциалы мышц, артефакты взаимодействия электродов с кожей, электронный шум усилителей и фоновый шум сети) [3–5]. Шумом принято считать высокочастотные компоненты кардиосигнала. Удаление шума приводит к сжатию и сглаживанию ЭКГ-сигнала.

Стадия выделения признаков из кардиосигнала представляет собой процесс извлечения требуемой информации (зубцы, комплексы и т. д.).

Рассмотрим подробнее каждую стадию.

Таблица 1. Свойства гауссовых и биортогональных вейвлетов

Критерий	Гауссовы вейвлеты (gaus)	Биортогональные вейвлеты (bior)
Наличие функция ϕ	–	+
Наличие функция ψ	+	+
	(явно выражена)	
Ортогональный анализ	–	+
Наличие компактного носителя	–	+
		($\phi, \psi, \tilde{\phi}, \tilde{\psi}$)
Возможность восстановления	Восстановление не гарантируется	+
Симметрия	+	+
FIR-фильтры	–	+
Возможный вейвлет-анализ	CWT без использования быстрых алгоритмов	CWT и DWT с использованием быстрых алгоритмов

Таблица 2. Точность обнаружения координат точек ЭКГ-сигнала

Критерий	Gaus	Bior		
		«bior1.1»	«bior1.3»	«bior1.5»
Точность обнаружения координат точек ЭКГ-сигнала, %	91–92	93–94	95–96	98–99

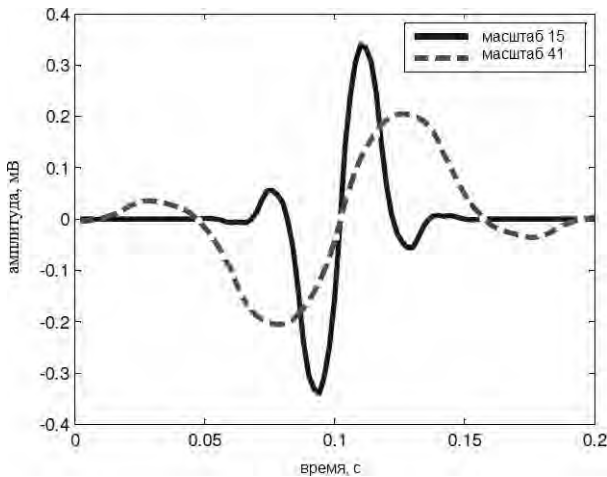


Рис. 2. Вейвлет «bior1.5» в масштабах 15 и 41



Рис. 3. Структура обработки ЭКГ-сигнала

2.1. Предварительная обработка

В простейшей модели предполагается, что зашумленный сигнал имеет вид (2) [4, 5]:

$$s(n) = f(n) + \sigma \cdot e(n), \quad (2)$$

где $f(n)$ – полезный сигнал, σ – уровень шума и $e(n)$ – гауссов белый шум, т. е. стационарная случайная последовательность с нулевым математическим ожиданием, абсолютно некоррелированная и с дисперсией, равной единице.

При вейвлет-анализе сигнал раскладывается на аппроксимирующие коэффициенты, которые представляют сглаженный сигнал, и детализирующие коэффициенты, описывающие колебания. Следовательно, шумовая компонента лучше отражается в детализирующих коэффициентах. Подобные составляющие могут быть удалены с использованием процедуры обнуления или пересчета коэффициентов детализации, значения которых меньше по сравнению со значением порога. Процедура пороговой обработки, или «трешолдинг», на сегодняшний день является пер-

спективным инструментом для «очистки» кардиосигнала от шумов (высоочастотных компонент) [4].

Существуют следующие виды трешолдинга [4, 5]:

1) жесткий трешолдинг, при котором все коэффициенты, превышающие некоторый порог, считаются принадлежащими к «оригинальному» сигналу, а остальные относят к шуму и обнуляют (3):

$$f(x) = \begin{cases} x, & |x| \geq t, \\ 0, & |x| < t, \end{cases} \quad (3)$$

где t – некоторый порог (коэффициент трешолдинга);

2) мягкий трешолдинг (4):

$$f(x) = \begin{cases} x - t, & x \geq t, \\ 0, & |x| < t, \\ x + t, & x \leq -t. \end{cases} \quad (4)$$

Качество шумоподавления сигнала и, следовательно, степень увеличения отношения сигнал/шум зависят не только от вида функции трешолдинга, но также от способа ее применения. В зависимости от этого трешолдинг делится на глобальный и локальный, а локальный в свою очередь на общий и многоуровневый [2, 3].

Для определения пороговых значений будем использовать следующие методы:

1) *SQR-LOG метод* (5) [12, 13]:

$$t = \sqrt{2 \left(\frac{(\text{median}\{c(i), i = 1 \dots n\})^2}{0,6745} \right) \ln(n)}, \quad (5)$$

где 0,6745 – оценка среднеквадратичного отклонения белого Гауссового шума, $c(i)$ – вейвлет-коэффициенты.

2) *Метод Берга – Массара* (6) [14]:

$$t = |c(z)|, \\ z = \arg \min \left[-\sum \{c^2(i), i < k\} + 2\sigma^2 k \left(a + \ln \left(\frac{n}{k} \right) \right) \right]; \\ k = 1 \dots n, \quad (6)$$

где σ^2 – дисперсия шума, a – параметр разреженности $a \geq 1$.

Параметр разреженности является ключевым в методе Берга – Массара, так как именно его величина, задаваемая исследователем, определяет степень подавления имеющихся в сигнале шумов.

В методе Берга – Массара определяются три интервала изменений значения параметра a , которые задают величину «штрафа»:

- «высокий», при $2,5 \leq a \leq 10$;
- «средний», при $1,5 \leq a \leq 2,5$;
- «низкий», при $1 \leq a \leq 1,5$.

3) Метод Штейна (7) [15]:

$$T_m = \arg \min_{t \geq 0} [SURE(W)],$$

$$SURE(W) = \sigma^2 - \frac{1}{N} \left(2\sigma^2 \cdot \#\{n: |W(m, n)|\} - \sum_{k=1}^L \min(|W(m, n)|)^2 \right), \quad (7)$$

где $W(m, n)$ – вейвлет-коэффициенты на уровне разложения m ; L – длина вектора вейвлет-коэффициентов $W(m, n)$ на уровне m ; σ^2 – дисперсия шума; $\#S$ – мощность множества S .

В работе [16] был выбран вейвлет из семейства Добеши в качестве базисного. Введем следующее ограничение: базовый вейвлет может быть применен к дискретному вейвлет-преобразованию, порядок высокочастотных и низкочастотных фильтров, предназначенных для выделения детализирующей и аппроксимирующей составляющих, не должен превышать 10 (из-за большой ресурсоемкости). Вейвлеты «db2» и «db4» удовлетворяют данным требованиям.

Считается, что верхняя граничная частота кардиосигнала, заметно влияющая на ее форму, не превышает 100 Гц [4]. Поэтому компоненты сигнала частоты выше 100 Гц можно удалить без существенного изменения формы сигнала. Исходя из этого, рассчитаем уровень разложения сигнала для вейвлетов «db2» и «db4». Вейвлет «db2» имеет центральную частоту $F_r = 0,6667$ Гц [4]. Поскольку $\Delta t = 1/1024$, то центральная частота первого уровня разложения равна $F_{r1} = 0,6667 \times 1024 = 682,70$ Гц, далее для второго уровня $F_{r2} = 341,35$ Гц, для третьего уровня $F_{r3} = 170,68$, для четвертого уровня $F_{r4} = 85,34$ Гц. Аналогично для вейвлета «db4» с центральной частотой $F_r = 0,7143$ Гц [15]: $F_{r1} = 734,30$ Гц, $F_{r2} = 367,15$ Гц, $F_{r3} = 183,57$ Гц, $F_{r4} = 91,8$ Гц. Таким образом, для удаления компонент кардиосигнала, чья частота выше 100 Гц, требуется использовать четвертый уровень разложения, при этом мы получаем сжатие сигнала в $2^4 = 16$ раз.

В качестве меры при выборе базисного вейвлета, применяемого для удаления шумов, будем использовать среднеквадратическую ошибку (8) [16]:

$$MSE(w, l) = \frac{1}{N} \sum_{i=1}^N (s(i) - \theta(i))^2, \quad (8)$$

где $s(i)$ – исходный сигнал, $\theta(i)$ – очищенный сигнал, w – вейвлет-функция, l – уровень разложения сигнала.

Результаты сравнения по видам и методам трешолдинга для вейвлетов «db2» и «db4» на четвертом уровне разложения с использованием локального многоуровневого трешолдинга представлены в табл. 3 и 4.

Таблица 3. Сравнение методов и типов вейвлет-трешолдинга для «db2»

	Жесткий трешолдинг	Мягкий трешолдинг
Метод Берга – Массара	0,0447	0,0563
SQR-LOG метод	2,5625	9,1393
Метод Штейна	1,0046	1,0239

Таблица 4. Сравнение методов и типов вейвлет-трешолдинга для «db4»

	Жесткий трешолдинг	Мягкий трешолдинг
Метод Берга – Массара	0,0393	0,0530
SQR-LOG метод	2,4873	8,9010
Метод Штейна	1,0073	1,0309

Согласно полученным результатам, наилучшим вейвлет-носителем для фильтрации сигнала от шумов является вейвлет «db4» на четвертом уровне разложения, а лучшим методом при локальном многоуровневом вейвлет-трешолдинге является жесткий трешолдинг с использованием метода Берга – Массара и штрафным порогом $a = 6$ (рис. 4, 5).

2.2. Выделение P-QRS-T комплексов

Как было сказано ранее, для выделения важных координат точек ЭКГ-сигнала (начало, пик и смещение Т-зубца, QRS-комплекса и Р-зубца) при непрерывном вейвлет-преобразовании требуется использование вейвлета «bior1.5» и 15 масштаб для обнаружения QRS-комплекса или 41 масштаб для обнаружения Р и Т зубцов. В работах [7–11] предложен следующий подход: выбираются две подряд пары коэффициентов, пересекающие нуль, между каждой из которых лежит локальный максимум (минимум) соответственно; в результате первый вейвлет-коэффициент является началом зубца, средний – пиком, а последний – смещением. Данный подход применяется для 15 и 41 масштабов в отдельности.

У предложенного выше метода низкое быстродействие (для использования в 24-часовом Холеровском мониторинге) из-за постоянного поиска локального максимума (минимума) в парах вейвлет-коэффициентов. Поэтому вместо поиска экстремумов авторами было предложено использовать пороговое значение, равное половине глобального максимума (минимума): $t_1 = 0,5 \max(\{C(i)\})$ или $t_2 = 0,5 \min(\{C(i)\})$, где $C(i)$ – вейвлет-коэффициенты, $i = 1 \dots N$. Также не учтена ситуация, когда имеется большая амплитудная разница между QRS-комплексом и Р, Т зубцами. В этом случае метод является малоэффективным. Поэтому, для точного обнаружения Р и Т зубцов, ав-

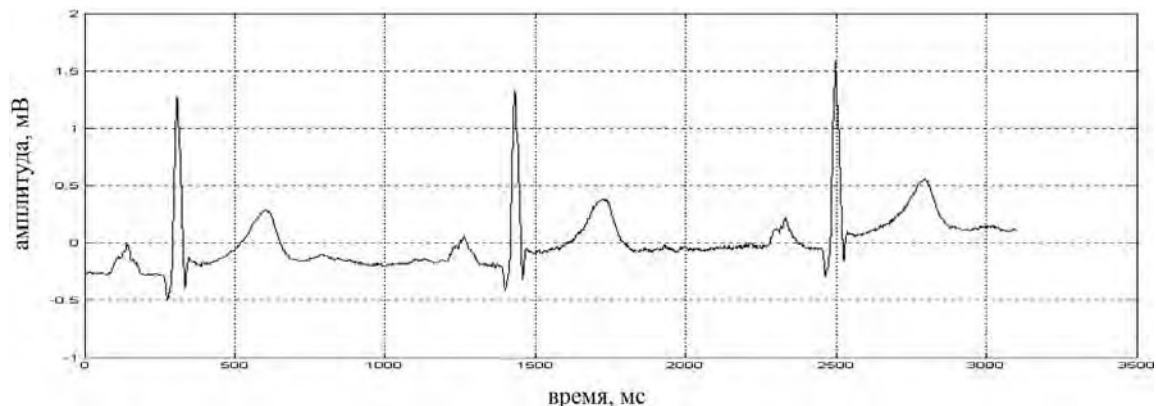


Рис. 4. Исходный кардиосигнал

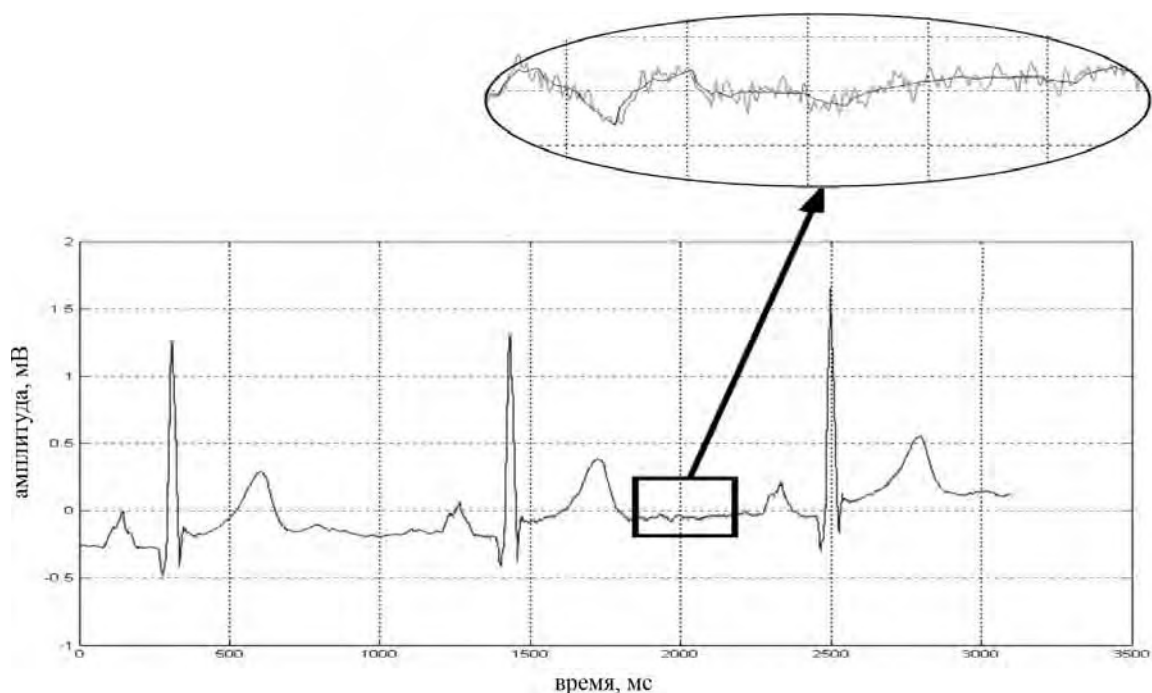


Рис. 5. Результат обработки ЭКГ-сигнала

торами предложено после обнаружения QRS-комплексов «удалить» их (QRS-комплексы), аппроксимируя линейно каждый такой участок сигнала.

Таким образом, процедура обнаружения важных координат точек кардиосигнала состоит из следующих этапов.

1. Выделение QRS-комплексов:

- Применить CWT с масштабным коэффициентом $a = 15$.

- Подсчитать пороговые значения $t_1 = 0,5\max(\{C(i)\})$ и $t_2 = 0,5\min(\{C(i)\})$.

- Найти пары коэффициентов, пересекающие нуль.

- Выбрать две последовательные пары так, чтобы между каждой из них находился вейвлет-коэффици-

ент, превышающий по модулю значение порога t_1 и t_2 . Тогда первый вейвлет-коэффициент будет началом QRS-комплекса, второй – пиком комплекса, а третий – его смещением.

2. «Удаление» QRS-комплексов:

Выполнить линейную аппроксимацию участков сигнала между точками, обозначенными как начало и смещение QRS-комплексов.

3. Выделение P и T зубцов:

- Применить CWT с масштабным коэффициентом $a = 41$.

- Подсчитать пороговые значения $t_1 = 0,5\max(\{C(i)\})$ и $t_2 = 0,5\min(\{C(i)\})$.

Найти пары коэффициентов, пересекающие нуль.

Выбрать две последовательные пары так, чтобы между каждой из них находился вейвлет-коэффициент, превышающий по модулю значение порога t_1 и t_2 . Тогда первый вейвлет-коэффициент будет началом R-зубца, второй – пиком зубца, а третий – его смещением.

Выбрать следующие две последовательные пары таким же образом, как и две предыдущие. Теперь первый вейвлет-коэффициент будет началом T-зубца, второй – пиком зубца, а третий – его смещением.

Этапы работы процедуры проиллюстрированы на рис. 6–10.

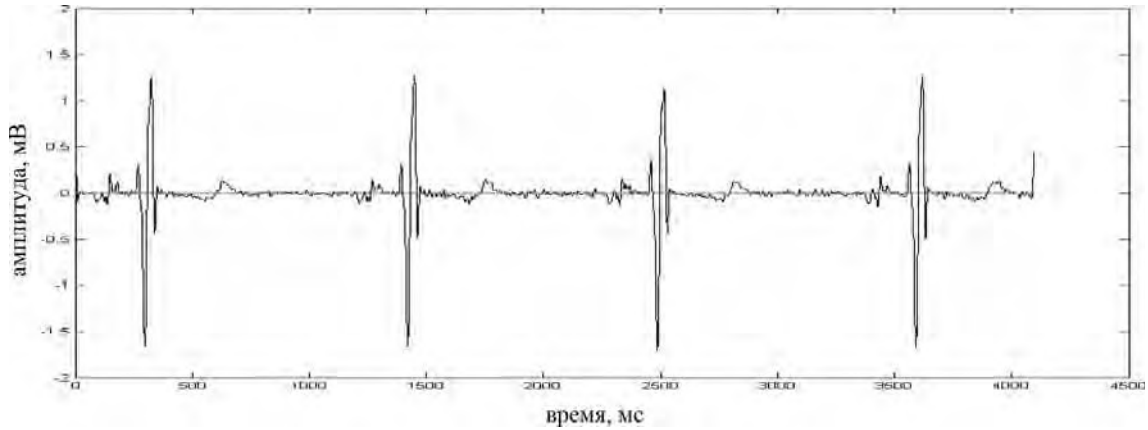


Рис. 6. Применение SWT с масштабным коэффициентом $a=15$ к ЭКГ-сигналу

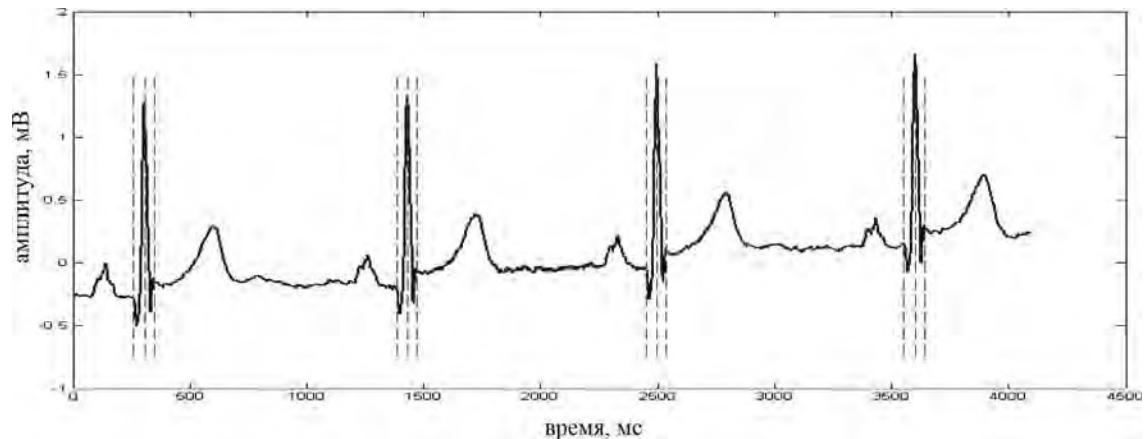


Рис. 7. Выделенные QRS-комплексы ЭКГ-сигнала

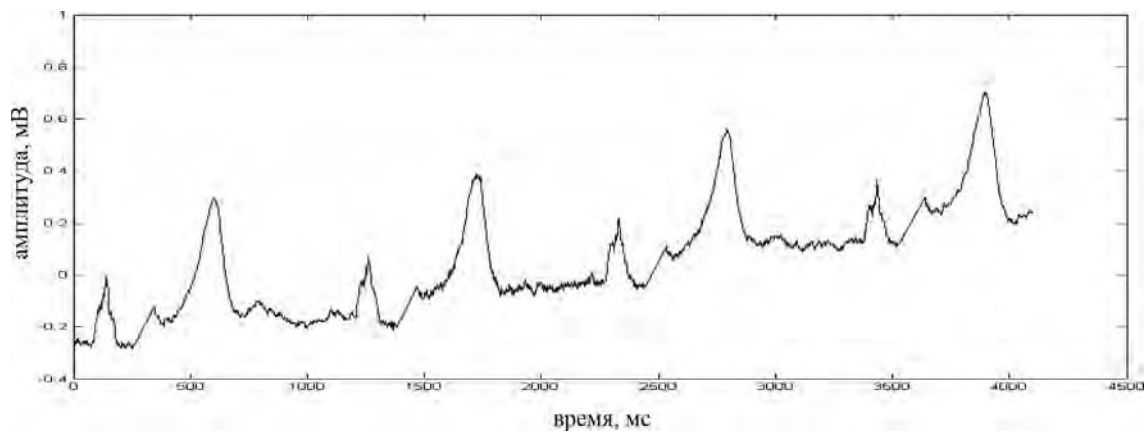


Рис. 8. «Удаление» QRS-комплексов из ЭКГ-сигнала

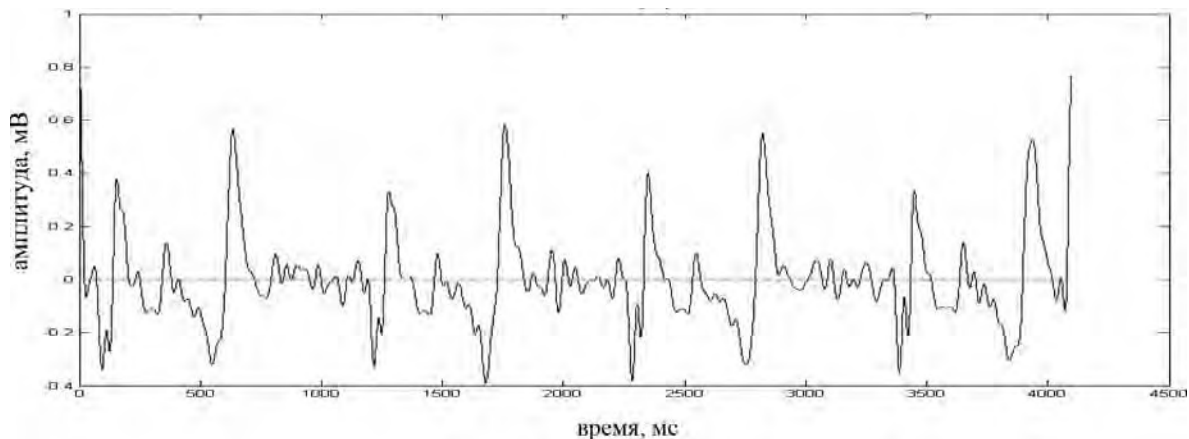


Рис. 9. Применение CWT с масштабным коэффициентом $a = 41$ к ЭКГ-сигналу

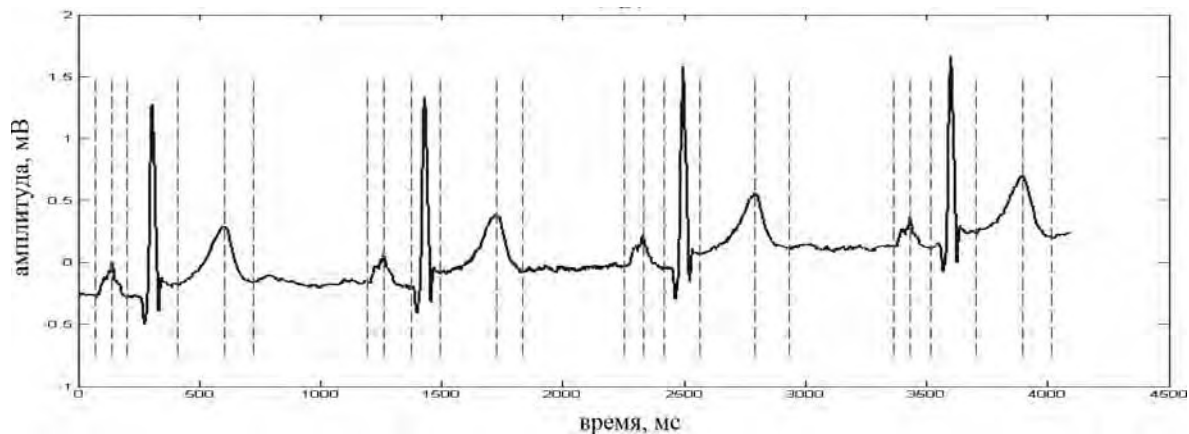


Рис. 10. Выделенные Р и Т зубцы ЭКГ-сигнала

3. ЭКСПЕРИМЕНТАЛЬНЫЕ РЕЗУЛЬТАТЫ

Для определения важных координат точек кардиосигнала (начало, пик и смещение Р-зубца, QRS-комплекса и Т-зубца), использовались базы данных ЭКГ Phisionet. Было протестировано тридцать пять кардиограмм людей с различными заболеваниями сердца

(30) и здоровых людей (5). Среди них: блокада пучков Гиса (6), атриовентрикулярная блокада (3), гипертрофия миокарда (6), стенокардия (5), инфаркт миокарда (10).

Результаты тестирования кардиосигналов здоровых людей и людей с различными патологиями приведены в табл. 5.

Таблица 5. Точность обнаружения P-QRS-T комплексов в ЭКГ-сигналах здоровых людей и людей с различными патологиями

Диагноз	Точность обнаружения, %
Блокада пучков Гиса	97–98
Атриовентрикулярная блокада	98–99
Гипертрофия миокарда	98–99
Стенокардия	98–99
Инфаркт миокарда	97–98
Без патологий	98–99

ВЫВОДЫ

1. Обоснован выбор типа вейвлет-преобразования и базисного вейвлета для анализа кардиосигналов (непрерывное вейвлет-преобразование с использованием базиса «*bio1.5*»).

2. Обоснован выбор масштабного коэффициента при непрерывном вейвлет-преобразовании для обнаружения P, QRS и T зубцов (15 масштаб для обнаружения QRS-комплекса и 41 масштаб для обнаружения P и T зубцов).

3. Усовершенствован метод обнаружения P-QRS-T зубцов: предложено пороговое значение и использование аппроксимации сигнала в областях QRS-комплекса с целью повышения точности обнаружения P и T зубцов.

4. Проанализированы существующие методики очистки сигналов от шумов.

5. Обоснован выбор типа и метода вейвлет-трешолдинга (локальный многоуровневый жесткий трешолдинг с использованием метода Берга – Массара и вейвлета «*db4*» на четвертом уровне разложения в качестве базиса).

Разработанный метод анализа ЭКГ-сигналов на основе вейвлет-преобразования в системе электрокардиографии высокого разрешения позволяет «очистить» кардиосигнал от шумов без потери информации. С помощью метода можно обнаружить девять важных координат точек кардиосигнала: начало, пик и смещение P-зубца, QRS-комплекса и T-зубца с точностью 98–99 %. Предлагаемый подход значительно увеличивает точность обнаружения P и T зубцов.

Направлением дальнейших исследований является разработка системы анализа и диагностики кардиосигнала.

СПИСОК ЛИТЕРАТУРЫ

1. Зудбинов, Ю. И. Азбука ЭКГ и боли в сердце / Ю. И. Зудбинов. – Ростов-на-Дону: Феникс, 2008. – 240 с.
2. Хан, М. Г. Быстрый анализ ЭКГ / М. Г. Хан. – М.: Бинном, 1999. – 230 с.
3. Хэмpton, Дж. Атлас ЭКГ: 150 клинических ситуаций / Дж. Хэмpton. – М.: Медицинская литература, 2007. – 320 с.
4. Смоленцев, Н. К. Основы теории вейвлетов. Вейвлеты в MATLAB / Смоленцев Н. К. – М.: ДМК Пресс, 2008. – 448 с.
5. Алексеев, К. А. Очерк «Вокруг CWT» [Электронный ресурс] / Алексеев К. А. – Электрон. дан. – Режим доступа: <http://matlab.exponenta.ru/wavelet/book3/index.php>, вільний. – Заг. з екрану.
6. Martinez, J. P. A wavelet-based ECG delineator: evaluation on standard databases / Martinez J. P., Almeida R., Laguna P. // IEEE Transactions on Biomedical Engineering. – 2004. – Vol. 51. – P. 570–581.
7. Vitec, M. A wavelet-based ECG delineation in Multilead ECG signals: Evaluation on the CSE Database / Vitec M.,

Hrubes J., Kozumplik J. // IFMBE Proceedings. – 2009. – Vol. 25. – P. 177–180.

8. Sahambi, J. S. Using wavelet transform for ECG characterization / Sahambi J. S., Tandon S. B. // IEEE Engineering in Medicine and Biology. – 2000. – Vol. 9. – P. 1532–1546.
9. Chouhan, V. S. Delineation of QRS-complex, P and T-wave in 12-lead ECG / Chouhan V. S., Mehta S. S., Lingayat N. S. // IJCSNS International Journal of Computer Science and Network Security. – 2008. – Vol. 8. – P. 185–190.
10. De Chazazl, P. Automatic measurement of the QRS onset and offset in individual ECG leads / De Chazazl P., Celler B. // IEEE Engineering in Medicine and Biology Society. – 1996. – Vol. 4. – P. 1399–1403.
11. Laguna, P. Automatic detection of wave boundaries in multilead ECG signals / Laguna P., Jane R., Caminal P. // Computers and Biomedical Research. – 1994. – Vol. 27. – P. 45–60.
12. Dohoto, D. L. De-Noising by soft-thresholding / Dohoto D. L. // IEEE Transactions on Information Theory. – 1995. – Vol. 41. – P. 613–627.
13. Dohoto, D. L. Ideal spatial adaptation via wavelet shrinkage / Dohoto D. L., Johnstone I. M. // Biometrika. – 1994. – Vol. 81. – P. 425–455.
14. Walden, A. T. Spectrum estimation by wavelet thresholding of multitaper estimators / Walden A. T., Percival D. B., McCoy E. J. // IEEE Transactions on Signal Processing. – 1998. – Vol. 46. – P. 3153–3165.
15. Luisier, F. A new SURE approach to image denoising: Interscale Orthonormal wavelet thresholding / Luisier F., Blu T., Unser M. // IEEE Transactions on image processing. – 2007. – Vol. 16. – P. 593–606.
16. Stein, C. Estimation of the mean of a multivariate normal distribution / Stein C. // Ann. Statist. – 1981. – Vol. 9. – P. 1135–1151.

Надійшла 14.09.2010

Дубровін В. І., Твердохліб Ю. В.

ВДОСКОНАЛЕННЯ МЕТОДІВ АНАЛІЗУ ЕКГ-СИГНАЛІВ НА ОСНОВІ ВЕЙВЛЕТ-ПЕРЕТВОРЕННЯ В СИСТЕМІ ЕЛЕКТРОКАРДІОГРАФІЇ ВИСОКОГО ПІДСИЛЕННЯ

У цій статті запропоновано модифікований метод аналізу ЕКГ-сигналів, який базується на безперервному вейвлет-перетворенні та одномасштабному підході. Алгоритм дозволяє знайти дев'ять важливих координат точок у всіх відділеннях кардіосигналу: початок, пік та зсув T зубця, QRS комплексу та P зубця. Реалізований алгоритм забезпечує високу точність у виявленні P і T зубців, що було невирішеною частиною попередніх методів. Також запропоновано оптимальний спосіб фільтрації кардіосигналу від шумових складових, який є модифікацією методів вейвлет-трешолдинга.

Ключові слова: електрокардіограма, P-QRS-T комплекси, вейвлет-перетворення.

Dubrovin V. I., Tverdohlib J. V.

IMPROVEMENT OF ECG SIGNALS ANALYSIS BASED ON WAVELET CONVERSION IN HIGH-RESOLUTION ELECTROCARDIOGRAPHY SYSTEM

In this paper we present an improved method of ECG signals delineation based on continuous wavelet conversion and single-scale approach. The algorithm permits to detect nine reference points in all ECG parts: onset, peak and offset of T wave, QRS complex and P wave. The implemented algorithm significantly increases the accuracy of P wave and T wave detection, which was the problem of the previous methods. We propose also the improved method of ECG signals filtering from noise as updating of the wavelet-thresholding methods.

Key words: electrocardiogram, P-QRS-T waves, wavelet conversion.

АДАПТИВНАЯ НЕЙРО-ФАЗЗИ СЕТЬ КОХОНЕНА

В статье предложен рекуррентный алгоритм обучения составной кластеризирующей нейро-фаззи сети Кохонена, являющейся обобщением WTA и WTM принципов обучения и алгоритмов Кашьяпа – Блейдона и Цыпкина, а также объединяющий в себе вероятностный и вероятностный подходы к кластеризации.

Ключевые слова: алгоритм самообучения, кластеризация, нейро-фаззи сеть, самоорганизующаяся карта Кохонена.

ВВЕДЕНИЕ

Задача кластеризации (классификации без учителя) достаточно часто встречается во многих приложениях, связанных с интеллектуальным анализом данных. Традиционный подход к решению этих задач предполагает, что каждое наблюдение может относиться только к одному кластеру, хотя более естественной представляется ситуация, когда обрабатываемый вектор признаков с разными уровнями принадлежности (вероятности, возможности) может принадлежать сразу нескольким классам. Данная ситуация является предметом рассмотрения нечеткого (фаззи) кластерного анализа, интенсивно развивающегося в двух направлениях: вероятностном [1–5] и возможностном [6, 7] подходах. Большинство алгоритмов нечеткой кластеризации предназначено для работы в пакетном режиме, когда весь массив данных, подлежащих обработке, задан априорно. В то же время существует широкий класс задач, когда данные поступают на обработку последовательно, в on-line режиме. Алгоритмов, предназначенных для решения этих задач, известно сравнительно немного [8–10], при этом они реализуют вероятностный подход на основе рекуррентной оптимизации принятой нечеткой целевой функции [1].

Для последовательной обработки данных при решении задачи кластеризации наилучшим образом приспособлены искусственные нейронные сети Кохонена [11], имеющие однослойную архитектуру с латеральными связями и обучаемые на основе принципов «победитель получает все» (WTA) или «победитель получает больше» (WTM). Данные сети продемонстрировали свою эффективность при решении многих задач с непересекающимися кластерами. Необходимость решения задач кластеризации в последовательном режиме обработки в условиях пересека-

ющихся классов привела к появлению самообучающихся гибридных нейро-фаззи систем, являющихся обобщением нейронной сети Кохонена и обладающих, благодаря использованию специальных алгоритмов настройки своих семантических весов, более широкими функциональными возможностями. Так, в [12, 13] была введена модификация сети Кохонена, основанная на нечетких правилах. Данная сеть показала свою эффективность в ряде задач, связанных с распознаванием образов, однако численная громоздкость затрудняет ее использование в on-line режиме. В [14] была предложена сеть Кохонена с нечетким выводом, обучаемая на основе комбинации правил Кохонена и Гроссберга. Основным недостатком этой конструкции является зависимость получаемых результатов от выбора свободных параметров процедуры обучения. В [15] была предложена, а в [16] развита, так называемая нечеткая кластеризирующая сеть Кохонена (FKCN), основанная на алгоритме нечетких c -средних (FCM) Бездека [1], который может быть записан и в рекуррентной форме. Лежащий в ее основе вероятностный подход к задаче кластеризации имеет принципиальные недостатки, ограничивающие ее использование в задачах с неизвестным заранее количеством кластеров.

В данной работе мы вводим новую двухслойную адаптивную нечеткую кластеризирующую сеть Кохонена (AFKCN), являющуюся адаптивной модификацией FKCN [15] и дальнейшим развитием конструкции, рассмотренной в [17], а также рекуррентные алгоритмы ее обучения, являющиеся обобщением правил Кохонена (WTA и WTM) и предназначенные для решения задачи в режиме последовательной обработки данных на основе как вероятностного, так и возможностного подходов.

1. АРХИТЕКТУРА АДАПТИВНОЙ НЕЧЕТКОЙ КЛАСТЕРИЗУЮЩЕЙ СЕТИ КОХОНЕНА

Архитектура предлагаемой нейро-фаззи сети приведена на рис. 1 и содержит два слоя: слой Кохонена, в котором определяются прототипы (центроиды) кластеров, и слой вычисления принадлежности.

Входные векторы-образы $x(k) = (x_1(k), x_2(k), \dots, x_n(k))^T$ (здесь $k = 1, 2, \dots, N, \dots$ имеет смысл или номера образа в обучающей выборке или текущего дискретного времени) с рецепторного (нулевого) слоя последовательно подаются на нейроны слоя Кохонена N_j^K , настраиваемые синаптические веса которого $c_{ji}(k)$, $j = 1, 2, \dots, m$; $i = 1, 2, \dots, n$ определяют центроиды m пересекающихся кластеров $c_j(k) = (c_{j1}(k), \dots, c_{jn}(k))^T$. В выходном слое, образованном нейронами N_j^M , вычисляются уровни принадлежности $u(k) = (u_1(k), u_2(k), \dots, u_m(k))^T$ текущего вектора $x(k)$ к j -му кластеру. По латеральным связям слоя Кохонена (показаны пунктиром) реализуются процессы конкуренции и кооперации, лежащие в основе WTM и WTA принципов.

2. АДАПТИВНЫЙ РЕКУРРЕНТНЫЙ АЛГОРИТМ ОБУЧЕНИЯ НЕЧЕТКОЙ КЛАСТЕРИЗАЦИИ

В классе процедур нечеткой кластеризации наиболее строгими с математической точки зрения являются алгоритмы, основанные на целевых функциях [1] и решающие задачу их оптимизации при тех или иных априорных предположениях. Наиболее распространенным здесь является вероятностный под-

ход, основанный на минимизации критерия (целевой функции)

$$E(u_j, c_j) = \sum_{k=1}^N \sum_{j=1}^m u_j^\beta \|x(k) - c_j\|^2 \quad (1)$$

при ограничениях

$$\sum_{j=1}^m u_j(k) = 1, \quad (2)$$

$$0 \leq \sum_{k=1}^N u_k(k) \leq N, \quad (3)$$

где $u_j(k) \in [0, 1]$, β – неотрицательный параметр фаззикации (*фаззификатор*), определяющий размытость границ между кластерами, $k = 1, 2, \dots, N$. Результатом кластеризации является $(N \times m)$ -матрица $U = \{u_j(k)\}$, называемая матрицей нечеткого разбиения.

Заметим, что поскольку элементы матрицы U могут рассматриваться как вероятности гипотез принадлежности векторов данных определенным кластерам, то процедуры, порождаемые минимизацией (1) при ограничениях (2), (3), называются вероятностными алгоритмами нечеткой кластеризации.

Вводя функцию Лагранжа

$$\begin{aligned} L(u_j(k), c_j, \lambda(k)) &= \\ &= \sum_{k=1}^N \sum_{j=1}^m u_j^\beta(k) \|x(k) - c_j\|^2 + \\ &+ \sum_{k=1}^N \lambda(k) \left(\sum_{j=1}^m u_j(k) - 1 \right) \end{aligned}$$

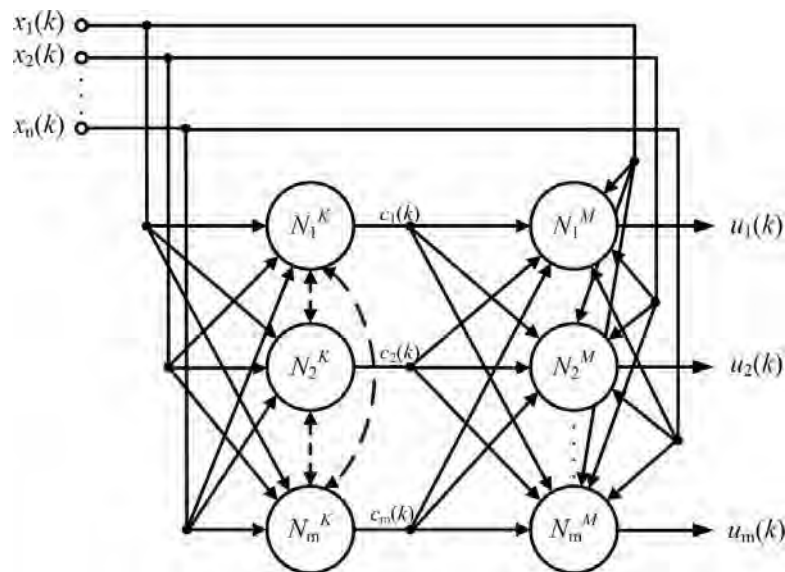


Рис. 1. Адаптивная нечеткая кластеризующая сеть Кохонена

(здесь $\lambda(k)$ – неопределенный множитель Лагранжа) и решая систему уравнений Каруша – Куна – Таккера, несложно получить искомое решение в виде

$$\left\{ \begin{aligned} u_j(k) &= \frac{(\|x(k) - c_j\|^2)^{\frac{1}{1-\beta}}}{\sum_{l=1}^m (\|x(k) - c_l\|^2)^{\frac{1}{1-\beta}}}, \\ c_j &= \frac{\sum_{k=1}^N u_j^\beta(k)x(k)}{\sum_{k=1}^N u_j^\beta(k)}, \\ \lambda(k) &= -\left(\left(\sum_{l=1}^m \beta \|x(k) - c_l\|^2\right)^{\frac{1}{1-\beta}}\right)^{1-\beta}, \end{aligned} \right. \quad (4)$$

совпадающее при $\beta = 2$ с алгоритмом нечетких С-средних Бездека:

$$\left\{ \begin{aligned} u_j(k) &= \frac{\|x(k) - c_j\|^2}{\sum_{l=1}^m \|x(k) - c_l\|^2}, \\ c_j &= \frac{\sum_{k=1}^N u_j^2(k)x(k)}{\sum_{k=1}^N u_j^2(k)}. \end{aligned} \right. \quad (5)$$

Использование процедур (4), (5) подразумевает, что вся выборка, подлежащая кластеризации и содержащая N наблюдений, задана заранее и не может изменяться в процессе обработки.

Для последовательной обработки данных, поступающих в on-line режиме, в [18–20] был использован рекуррентный алгоритм, в основе которого лежит процедура нелинейного программирования Эрроу – Гурвица – Удзавы:

$$\left\{ \begin{aligned} u_j(k+1) &= \frac{(\|x(k+1) - c_j(k)\|)^{\frac{2}{1-\beta}}}{\sum_{l=1}^m (\|x(k+1) - c_l(k)\|)^{\frac{2}{1-\beta}}}, \\ c_j(k+1) &= c_j(k) + \eta(k)u_j^\beta(k+1)(x(k+1) - c_j(k)), \end{aligned} \right. \quad (6)$$

являющийся обобщением алгоритма обучения Чанга – Ли [9, 10] и совпадающий при $\beta = 2$ с градиентной процедурой кластеризации Парка – Дэггера [8]

$$\left\{ \begin{aligned} u_j(k+1) &= \frac{(\|x(k+1) - c_j(k)\|)^{-2}}{\sum_{l=1}^m (\|x(k+1) - c_l(k)\|)^{-2}}, \\ c_j(k+1) &= c_j(k) + \eta(k)u_j^2(k+1)(x(k+1) - c_j(k)). \end{aligned} \right.$$

Рассматривая соотношения (6) с позиций обучения нейросети Кохонена, можно заметить, что множитель $u_j^\beta(k+1)$ во втором выражении соответствует функции соседства в правиле обучения на основе WTM-принципа.

Несложно видеть, что при $\beta = 1$ и $u_j(k) \in \{0, 1\}$ процедура (6) совпадает с четким алгоритмом С-средних (НСМ), а при $\beta = 0$ получаем стандартное WTA-правило обучения Кохонена [11]

$$c_j(k+1) = c_j(k) + \eta(k)(x(k+1) - c_j(k)), \quad (7)$$

первоначально введенное Кашьяпом – Блейдоном [21] в шестидесятых годах прошлого века. Правило (7), как можно заметить, оптимизирует целевую функцию

$$E(c_j) = \sum_{k=1}^N \|x(k) - c_j\|^2, \quad \sum_{j=1}^m N_j = N,$$

прямая минимизация которой ведет к обычной оценке среднего арифметического

$$c_j(k) = \frac{1}{N_j} \sum_{k=1}^{N_j} x(k), \quad (8)$$

где N_j – число векторов $x(k)$, отнесенных к j -му кластеру в процессе конкуренции. Перепиывая (8) в рекуррентной форме, приходим к оптимальному алгоритму самообучения Цыпкина [22]

$$c_j(k+1) = c_j(k) + \frac{1}{N_j(k+1)} \sum_{k=1}^{N_j} (x(k+1) - c_j(k)),$$

(здесь $N_j(k+1)$ – число векторов $x(k)$, отнесенных к j -му кластеру на $(k+1)$ -й итерации алгоритма), являющемуся процедурой стохастической аппроксимации в чистом виде.

Таким образом, алгоритм (6) может рассматриваться в качестве правила обучения нейро-фаззи-сети, приведенной на рис. 1, объединяющего в себе вероятностную нечеткую кластеризацию, правила самообучения Кохонена, алгоритмы Кашьяпа – Блейдона и Цыпкина.

Основные недостатки вероятностного подхода связаны с ограничением (2), требующим равенства единице суммы принадлежностей конкретного вектора-образа всем кластерам. В простейшем случае двух кластеров и двух образов вектор, равноправно принадлежащий обоим классам, и вектор, не принадлежащий ни одному из них, но лежащий на равном расстоянии от обоих центроидов, имеют одинаковые уровни принадлежности, равные 0,5. Естественно, что данное обстоятельство, требующее точного априорного знания числа кластеров m и ухудшающее точность получаемых результатов, привело к появлению возможностного подхода к нечеткой кластеризации [6, 7].

В возможностных алгоритмах кластеризации целевая функция имеет вид

$$E(u_j, c_j) = \sum_{k=1}^N \sum_{j=1}^m u_j^\beta(k) \|x(k) - c_j\|^2 + \sum_{j=1}^m \mu_j \sum_{k=1}^N (1 - u_j(k))^\beta, \quad (9)$$

где скалярный параметр $\mu_j > 0$ определяет расстояние, на котором уровень принадлежности принимает значение 0,5, т. е. если

$$\|x(k) - c_j\|^2 = \mu_j,$$

то $u_j(k) = 0,5$.

Прямая оптимизация (9) по u_j, c_j дает очевидное решение

$$\left\{ \begin{aligned} u_j(k) &= \left(1 + \left(\frac{\|x(k) - c_j\|^2}{\mu_j} \right)^{\frac{1}{1-\beta}} \right)^{-1}, \\ c_j &= \frac{\sum_{k=1}^N u_j^\beta(k) x(k)}{\sum_{k=1}^N u_j^\beta(k)}, \\ \mu_j &= \frac{\sum_{k=1}^N u_j^\beta(k) \|x(k) - c_j\|^2}{\sum_{k=1}^N u_j^\beta(k)}, \end{aligned} \right. \quad (10)$$

при $\beta = 2$ принимающее вид

$$\left\{ \begin{aligned} u_j(k) &= \frac{\mu_j}{\mu_j + \|x(k) - c_j\|^2}, \\ c_j &= \frac{\sum_{k=1}^N u_j^2(k) x(k)}{\sum_{k=1}^N u_j^2(k)}, \\ \mu_j &= \frac{\sum_{k=1}^N u_j^2(k) \|x(k) - c_j\|^2}{\sum_{k=1}^N u_j^2(k)}. \end{aligned} \right. \quad (11)$$

Интересно заметить, что выражения, описывающие центроиды, в (4) и (10) и в (5) и (11) совпадают, однако принадлежности вычисляются различными способами.

Если данные поступают на обработку последовательно, то вместо (10), (11) могут быть использованы рекуррентные алгоритмы [17–20] вида

$$\left\{ \begin{aligned} u_j(k) &= \frac{1}{1 + \left(\frac{\|x(k) - c_j\|^2}{\mu_j(k)} \right)^{\frac{1}{1-\beta}}}, \\ c_j(k+1) &= c_j(k) + \eta(k) u_j^\beta(k+1) (x(k+1) - c_j(k)), \\ \mu_j(k+1) &= \frac{\sum_{p=1}^{k+1} u_j^\beta(p) \|x(p) - c_j(k+1)\|^2}{\sum_{p=1}^{k+1} u_j^\beta(p)} \end{aligned} \right. \quad (12)$$

и

$$\left\{ \begin{aligned} u_j(k) &= \frac{\mu_j(k)}{\mu_j(k) + \|x(k+1) - c_j(k)\|^2}, \\ c_j(k+1) &= c_j(k) + \eta(k) u_j^\beta(k+1) (x(k+1) - c_j(k)), \\ \mu_j(k+1) &= \frac{\sum_{p=1}^{k+1} u_j^\beta(p) \|x(p) - c_j(k+1)\|^2}{\sum_{p=1}^{k+1} u_j^\beta(p)}. \end{aligned} \right. \quad (13)$$

Несложно заметить, что процедуры (12), (13) также по сути являются WTM-правилами обучения сети Кохонена, однако, что интересно, первое соотношение, задающее уровень принадлежности образа к кластеру, есть стандартная колоколообразная функция принадлежности с изменяющимся во времени параметром ширины.

Алгоритмы, построенные на основе возможностного подхода, в отличие от вероятностных алгоритмов, позволяют в процессе обработки данных обнаруживать появление новых кластеров. Так, если уровень принадлежности наблюдения $x(k+1)$ ко всем кластерам окажется ниже некоторого наперед заданного порога, можно говорить о появлении $(m+1)$ -го кластера с начальными координатами центроида $c_{m+1} = x(k+1)$.

3. СОСТАВНАЯ АДАПТИВНАЯ НЕЧЕТКАЯ КЛАСТЕРИЗУЮЩАЯ СЕТЬ КОХОНЕНА (САФКСН)

Поскольку заранее невозможно сказать, какой из подходов – вероятностный или возможностный – более предпочтителен при решении конкретной задачи, целесообразно их одновременное использование. Кроме того, решения, полученные с помощью вероятностных процедур, рекомендуется использовать в качестве начальных условий для возможностных алгоритмов [7, 23]. В связи с этим целесообразно параллельное использование обоих подходов в рамках составной адаптивной нечеткой кластеризующей сети Кохонена (САФКСН), приведенной на рис. 2.

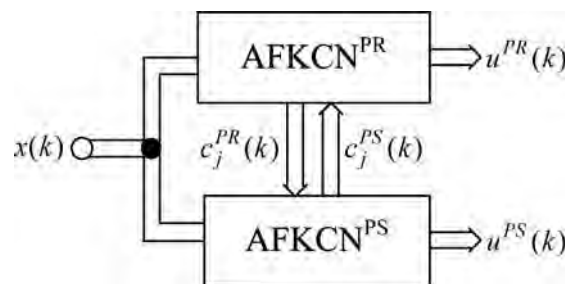


Рис. 2. Составная адаптивная нечеткая кластеризующая сеть Кохонена

Данная конструкция образована двумя параллельно включенными адаптивными нечеткими кластеризующими сетями Кохонена (AFKCN), одна из которых обучается с помощью алгоритма (6), а другая – алгоритма (12). Индексы PR и PS обозначают возможностную и вероятностную оценки соответственно.

Алгоритм обучения SAFKCN имеет вид

$$\left\{ \begin{array}{l} c_j^{PR}(k+1) = c_j^{PS}(k) + \eta(k)(u_j^{PR}(k))^\beta (x(k+1) - c_j^{PS}(k)), \\ u_j^{PR}(k+1) = \frac{\|x(k+1) - c_j^{PR}(k+1)\|^{\frac{2}{1-\beta}}}{\sum_{i=1}^m \|x(k+1) - c_i^{PR}(k+1)\|^{\frac{2}{1-\beta}}}, \\ c_j^{PS}(k+1) = c_j^{PR}(k) + \eta(k)(u_j^{PS}(k))^\beta (x(k+1) - c_j^{PR}(k)), \\ u_j^{PS}(k+1) = \frac{1}{1 + \left(\frac{\|x(k) - c_j^{PS}(k+1)\|^2}{\mu_j(k)} \right)^{\frac{1}{1-\beta}}}, \\ \mu_j(k+1) = \frac{\sum_{p=1}^{k+1} (u_j^{PS}(p))^\beta \|x(p) - c_j^{PS}(k+1)\|^2}{\sum_{p=1}^{k+1} (u_j^{PS}(p))^\beta}, \end{array} \right. \quad (14)$$

при этом при вычислении центроидов кластеров сети обмениваются между собой соответствующей информацией. Признаком правильного оценивания с помощью алгоритма (14) является выполнение неравенства

$$\sum_{l=1}^m \|c_l^{PR} - c_l^{PS}\|^2 \leq \varepsilon, \quad (15)$$

где ε определяет приемлемую точность кластеризации.

ВЫВОДЫ

В статье предлагается подход к обучению кластеризующей сети Кохонена в условиях пересекающихся кластеров на основе адаптивных процедур вероятностной и возможностной нечеткой кластеризации, являющихся обобщением WTA и WTM правил Кохонена, алгоритмов Кашьяпа – Блейдона и Цыпкина. Компьютерное моделирование демонстрирует эффективность развиваемого подхода для решения задач кластеризации данных в условиях неопределенности. Результаты могут быть использованы для решения задач интеллектуального анализа данных, ранней диагностики неисправностей, Web-Mining и т. п., когда размер выборки данных заранее неизвестен, а сами эти данные должны обрабатываться в реальном времени.

СПИСОК ЛИТЕРАТУРЫ

1. *Bezdek, J. C.* Pattern Recognition with Fuzzy Objective Function Algorithms / Bezdek J. C. – N. Y. : Plenum Press, 1981. – 272 p.

2. *Gath, I.* Unsupervised optimal fuzzy clustering / Gath I., Geva A. B. // Pattern Analysis and Machine Intelligence. – 1989. – 2. – 7. – P. 773–787.
3. *Höppner, F.* Fuzzy-Clusteranalyse. / Höppner F., Klawonn F., Kruse R. – Braunschweig : Vieweg, 1997. – 280 S.
4. *Höppner, F.* Fuzzy Clustering Analysis: Methods for Classification, Data Analysis and Image Recognition. / Höppner F., Klawonn F., Kruse R., Runkler T. – Chichester : John Willey & Sons, 1999. – 289 p.
5. *Gan, G.* Data Clustering: Theory, Algorithms and Applications / Gan G.; Ma Ch.; Wu J. – Philadelphia : SIAM, 2007. – 466 p.
6. *Krishnapuram, R.* A possibilistic approach to clustering / Krishnapuram R., Keller J. M. // Fuzzy Systems. – 1993. – 1. – № 2. – P. 98–110.
7. *Krishnapuram, R.* Fuzzy and possibilistic clustering methods for computer vision / Krishnapuram R., Keller J. M. // Neural Fuzzy Systems. – 1994. – 12. – P. 133–159.
8. *Park, D. C.* Gradient based fuzzy c-means (GBFCM) algorithm / Park D. C., Dagher I. // Proc. IEEE Int. Conf. on Neural Networks. – 1984. – P. 1626–1631.
9. *Chung, F. L.* Fuzzy competitive learning / Chung F. L., Lee T. // Neural Networks. – 1994. – 7. – № 3. – P. 539–552.
10. *Chung, F.-L.* Unsupervised fuzzy competitive learning with monotonically decreasing fuzziness / Chung F.-L., Lee T. // Proc. 1993 Int. Joint Conf. on Neural Networks. – 1993. – P. 2929–2932.
11. *Kohonen, T.* Self-Organizing Maps / Kohonen T. – Berlin : Springer-Verlag, – 1995. – 362 p.
12. *Vuorimaa, P.* Fuzzy self-organizing map / Vuorimaa P. // Fuzzy Sets and Systems. – 1994. – 66. – P. 223–231.
13. *Vuorimaa, P.* Use of the fuzzy self-organizing map in pattern self-recognition / Vuorimaa P. // Proc. 3-rd IEEE Int. Conf. Fuzzy Systems «FUZZ-IEEE'94». – Orlando, USA, 1994. – P. 798–801.
14. *Bodyanskiy, Ye.* Combined learning algorithm for a self-organizing map with fuzzy inference / Bodyanskiy Ye., Gorshkov Ye., Kolodyazhnyy V., Stephan A. // Computational intelligence, theory and applications: International Conference 8th Fuzzy Days in Dortmund, Germany, Sept. 29 – Oct. 01, 2004 ; proceedings / B. Reusch (Ed). – Berlin Heidelberg : Springer-Verlag, 2005. – P. 641–650. – (Advanced in Soft Computing ; Vol. 33).
15. *Gorshkov, Ye.* New recursive learning algorithms for fuzzy Kohonen clustering network / Gorshkov Ye., Kolodyazhnyy V., Bodyanskiy Ye. // Proc. 17th Int. Workshop on Nonlinear Dynamics of Electronic Systems. – Rapperswil, Switzerland, 2009. – P. 58–61.
16. *Pascual-Marqui, R. D.* Smoothly distributed fuzzy c-means: a new self-organizing map. / Pascual-Marqui R. D., Pascual Montano A. D., Kochi K., Carazo J. M. // Pattern Recognition. – 2001. – 34. – P. 2395–2402.
17. *Tsao, E. C.-K.* Fuzzy Kohonen clustering network / Tsao E. C.-K., Bezdek J. C., Pal N. P. // Pattern Recognition. – 1994. – 27. – № 5. – P. 757–764.
18. *Бодянский, Е. В.* Об адаптивном алгоритме нечеткой кластеризации данных / Бодянский Е. В., Горшков Е. В., Кокшенев И. В., Колодяжный В. В. // Адаптивные системы автоматического управления. – Вип. 5(25). – Дніпропетровськ : Системні технології, 2002. – С. 108–117.
19. *Bodyanskiy, Ye.* Recursive fuzzy clustering algorithms / Bodyanskiy Ye., Kolodyazhnyy V., Stephan A. // Proc. 10th East West Fuzzy Colloquium. – Zittau, Germany, 2002. – P. 276–283.
20. *Bodyanskiy, Ye.* Computational intelligence techniques for data analysis / Bodyanskiy Ye. // Lecture Notes in Informatics. – 2005. – Vol. P-72. – P. 15–36.
21. *Kashyap, R. L.* Estimation of probability density and distribution functions / Kashyap R. L., Blyaydon C. C. // IEEE Trans. on Information Theory. – 1968. – 14. – № 4. – P. 549–556.
22. *Цыпкин, Я. З.* Основы теории обучающихся систем / Цыпкин Я. З. – Москва : Наука, 1970. – 252 с.

23. Klawonn, F. Constructing a fuzzy controller from data / Klawonn F., Kruse R. // Fuzzy Sets and Systems. – 1997. – 85. – P. 117–193.

Надійшла 21.10.2010

Колчигін Б. В., Волкова В. В., Бодяньський С. В.
АДАПТИВНА НЕЙРО-ФАЗЗИ МЕРЕЖА КОХОНЕНА

У роботі запропоновано рекурентний алгоритм навчання складової нейро-фаззи мережі Кохонена для кластеризації. Запропонований алгоритм є узагальненням WTA, WTM принципів навчання та алгоритмів Кашьяпа-Блейдона та Ципкіна, а також об'єднує в собі імовірнісний та можливістьний методи навчання.

УДК 004.93

Субботин С. А.

Канд. техн. наук, доцент Запорозького національного технічного університета

ЭКСПЕРИМЕНТАЛЬНОЕ ИССЛЕДОВАНИЕ И АНАЛИЗ ВЗАИМОСВЯЗЕЙ ИНФОРМАЦИОННЫХ ПОКАЗАТЕЛЕЙ КАЧЕСТВА ДИАГНОСТИЧЕСКИХ НЕЙРОМОДЕЛЕЙ

Экспериментально исследован комплекс информационных показателей качества диагностических нейромоделей при решении практических задач. Получены качественные зависимости между показателями, учет которых при формировании интегральных показателей качества позволяет существенно упростить вычисления путем исключения дублирующихся показателей.

Ключевые слова: качество, модель, техническая диагностика, нейронная сеть, нейро-нечеткая сеть.

ВВЕДЕНИЕ

Необходимость обеспечения высокого уровня качества выпускаемых и эксплуатируемых изделий обуславливает потребность в разработке систем неразрушающей диагностики и контроля качества, для построения которых, в свою очередь, необходимо обладать диагностической моделью [1].

Одними из наиболее перспективных средств разработки диагностических моделей являются искусственные нейронные [2] и нейро-нечеткие сети [3], обладающие способностями к индуктивному обучению на основе прецедентов. В настоящее время известно достаточно большое количество различных архитектур нейронных и нейро-нечетких сетей, а также методов их обучения [2, 3]. Кроме того, даже для сети с фиксированной архитектурой и методом обучения вследствие свойства вариативности получаемых моделей по одной и той же выборке прецедентов возможно получить множество различных моделей, обладающих приемлемым уровнем точности. Поэтому на практике при разработке систем технической диагностики в нейросетевом или нейро-не-

Ключові слова: алгоритм самонавчання, кластеризація, нейро-фаззи мережа, самоорганізована карта Кохонена.

Kolchygin B., Volkova V., Bodyanskiy Ye.
ADAPTIVE NEURO-FUZZY KOHONEN'S NETWORK

Recurrent learning algorithm for compartmental adaptive fuzzy Kohonen clustering network is proposed. The algorithm is generalization of WTA and WTM concepts and Kashyap-Blaydon and Tsytkin learning algorithms and also combines probabilistic and possibilistic clustering methods.

Key words: clustering, neuro-fuzzy network, self-learning algorithm, self-organizing Kohonen map.

четком базисе является актуальной задача выбора одной из множества возможных диагностических моделей, что требует разработки информационных показателей, характеризующих качество модели.

Цель данной работы заключается в создании и экспериментальном исследовании комплекса показателей качества нейромоделей для автоматизации их построения и анализа.

1. ПОСТАНОВКА ЗАДАЧИ ОЦЕНКИ КАЧЕСТВА НЕЙРОМОДЕЛИ

Пусть задана обучающая выборка $\langle x, y \rangle$, $x = \{x^s\}$, $x^s = \{x_j^s\}$, $y = \{y^s\}$, $s = 1, 2, \dots, S$, $j = 1, 2, \dots, N$, где x_j^s – значение j -го входного признака s -го экземпляра x^s выборки, y^s – значение выходного признака, сопоставленное s -му экземпляру выборки, S – объем выборки, N – число признаков в выборке. На основе обучающей выборки $\langle x, y \rangle$ может быть получена нейромодель $net(w, x)$, где $w = \{w_j^{(\eta, i)}\}$, $w_j^{(\eta, i)}$ – весовой коэффициент, сопоставленный j -му входу i -го нейрона η -го слоя $x_j^{(\eta, i)}$, $net = \{\langle \varphi^{(\eta, i)}(w^{(\eta, i)}, x^{(\eta, i)}), \psi^{(\eta, i)}(w^{(\eta, i)}, x^{(\eta, i)}) \rangle\}$, $w^{(\eta, i)} = \{w_j^{(\eta, i)}\}$, $x^{(\eta, i)} = \{x_j^{(\eta, i)}\}$,

© Субботин С. А., 2011

$\varphi^{(\eta, i)}(w^{(\eta, i)}, x^{(\eta, i)})$ и $\psi^{(\eta, i)}(w^{(\eta, i)}, x^{(\eta, i)})$ – соответственно, весовая (дискриминантная) и активационная функции i -го нейрона η -го слоя.

Обозначим: M – число слоев в сети; N_η – число нейронов в η -м слое; N_n – число нейронов в сети; N_n^{\max} – максимально допустимое число нейронов; N_w – число весов в сети; $w_{j \max}^{(\eta, i)}$, $w_{j \min}^{(\eta, i)}$ – максимальное и минимальное возможные значения веса $w_j^{(\eta, i)}$; $w_{i, j}^{\max}$, $w_{i, j}^{\min}$ – максимальное и минимальное возможные значения веса связи от i -го нейрона к j -му; w_{\max} , w_{\min} – максимальное и минимальное возможные значения весов сети; $\psi_{\max}^{(M, i)}$, $\psi_{\min}^{(M, i)}$ – максимальное и минимальное значения i -го выхода.

Необходимо разработать информационные показатели $\{I_i\}$, характеризующие качество нейромодели $f(w, x)$, т. е. выражающие количественно наличие у модели тех или иных свойств, важных для пользователя.

2. ИНФОРМАЦИОННЫЕ ПОКАЗАТЕЛИ КАЧЕСТВА ДИАГНОСТИЧЕСКОЙ НЕЙРОМОДЕЛИ

Традиционно используемым показателем качества нейромодели является ее ошибка E (как правило, среднеквадратическая) [2]. Однако ошибка не отражает множество свойств, которыми обладают нейромодели.

В ранних работах [4, 5] автором была предпринята попытка разработки набора показателей, характеризующих логическую прозрачность и сложность нейронных сетей прямого распространения. Однако весьма значительный набор свойств нейросетей и нейро-нечетких сетей оставался неопианным аналитически выражающими его показателями.

Поэтому в последнее время в работах [6–9] предложены показатели: предорганизации в обучении I_{po} , обобщения I_G , соответствия нелинейностей выборки и нейромодели: \tilde{I}_{nl} , автоматизации в обучении I_{aut} , автоматизации предорганизации в обучении I_{apo} , адаптивности I_{adapt} , асимметрии весов I_{asym}^w , асимметрии модели I_{asym} , асимметрии нейронов I_{asym}^n , вариативности I_v , иерархичности I_h , интерпретируемости I_{interp} , максимальной надежности весов I_{wre}^{\max} , максимальной надежности нейронов $I_{n.re}^{\max}$, нейронной пластичности I_{np} , нелинейности модели $I_{nl}(net)$, обобщения I_{gen} , однородности активационных функций I_{ha} , однородности дискриминантных функций I_{hd} , однородности нейронов I_{hn} , параллелизма I_{par} , обучаемости I_l , пластичности нейромодели I_{pl} , разнообразия модели по отношению к генеральной совокупности $I_{div}(net, X, Y)$, разнообразия модели по отношению к обучающей выборке $I_{div}(net, \langle x, y \rangle)$,

разнообразия нейромодели $I_{div}(net)$, распределенности памяти I_{md} , робастности по отношению ко входам I_{rb}^x , робастности I_{rb} , робастности по отношению к весам I_{rb}^w , симметрии весов I_{sym}^w , симметрии модели I_{sym} , симметрии нейронов I_{sym}^n , синаптической пластичности I_{sp} , сложности сети L_{net} , средней надежности весов I_{wre}^{avg} , средней надежности нейронов $I_{n.re}^{avg}$, уверенности i -го выхода $I_{cert}(x^s)$, уверенности модели I_{cert} , функциональной блочности I_{fb} , устойчивости к шуму I_{tol}^{ξ} , чувствительности I_{tol}^{ξ} , эквивалентности I_{eq} , эквивалентности I_{eqf} , эластичности выходной переменной y_i по признаку x_j $El_{x_j}(net, y_i)$, эмерджентности I_{α} , эффективности модели I_{ef} , интегральный информационный критерий $ИС$. Данный комплекс показателей может быть дополнен информационными показателями SSD , $CAIC$, MDL , CP , AIC , $AICc$, HQC , BIC , $AICu$.

Совокупность предложенных показателей [6–9] позволяет в комплексе охарактеризовать качество диагностической нейромодели.

Поскольку количество разработанных показателей [6–9] достаточно велико, то весьма важным с практической точки зрения является иерархическое упорядочивание вычисления показателей при их совместном применении.

На рис. 1 приведен граф иерархии аналитических связей разработанных показателей. Применение данного графа при разработке программ позволяет существенно сократить объем вычислений за счет исключения повторов однотипных расчетов при расчете значительной части показателей качества.

Поскольку на качество нейромодели существенное влияние оказывает обучающая выборка данных, в работах [10–12] были предложены показатели качества выборки. На рис. 1 отображены аналитические зависимости показателей качества нейромодели от некоторых характеристик обучающей выборки: сложности L и нелинейности $I_{nl}(\langle x, y \rangle)$. Взаимосвязи между показателями качества выборки описаны в [11].

3. ЭКСПЕРИМЕНТЫ ПО ИССЛЕДОВАНИЮ ВЗАИМОСВЯЗИ ПОКАЗАТЕЛЕЙ КАЧЕСТВА

Расчет комплекса предложенных показателей качества, даже с оптимизацией вычислений путем использования графа, приведенного на рисунке, может оказаться на практике вычислительно затратным. Поэтому представляет существенный интерес исследование взаимосвязей между показателями качества с целью устранения из комплексных показателей избыточных вычислений.

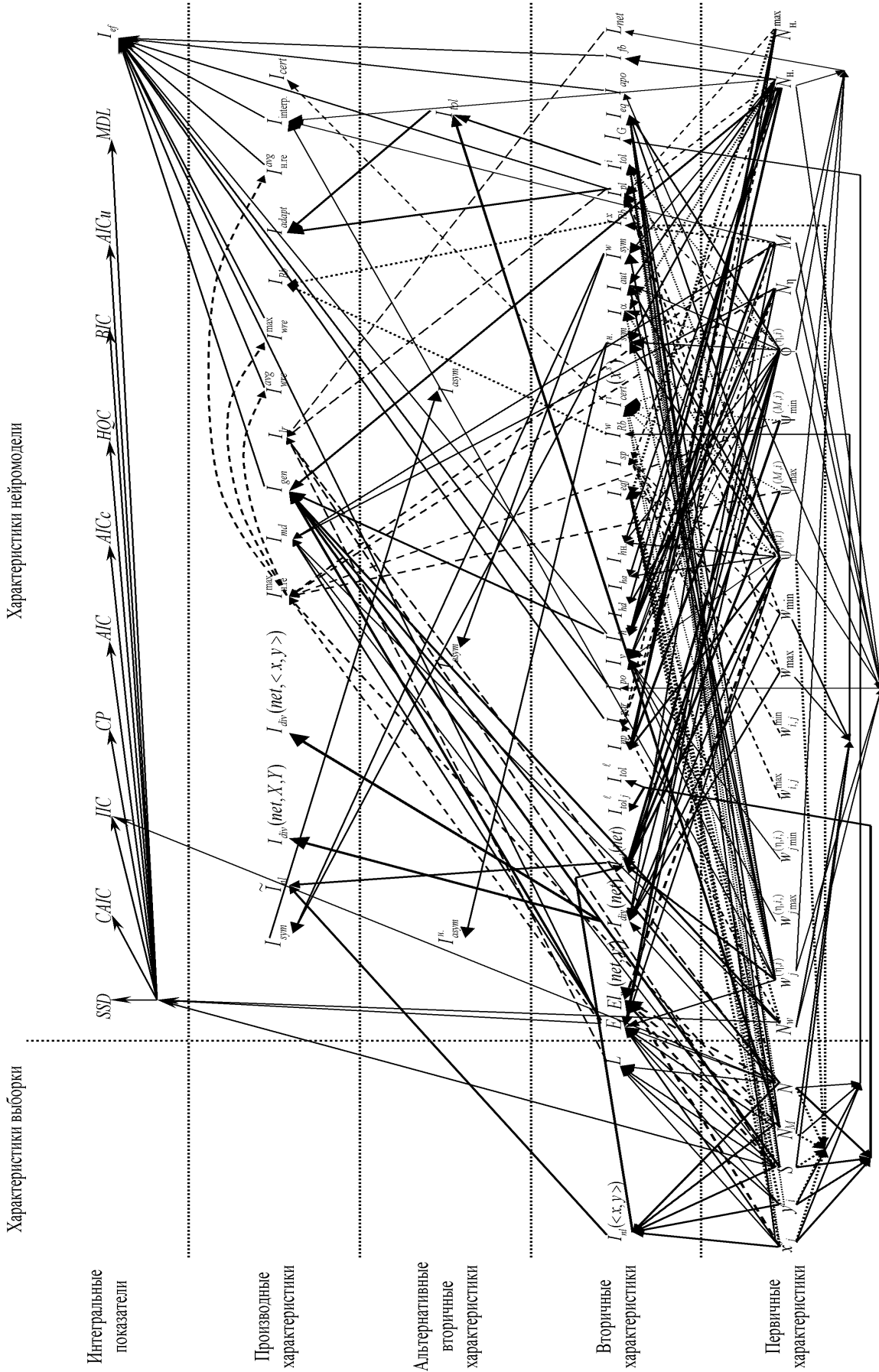


Рис. 1. Схема аналитических взаимосвязей информационных показателей качества диагностических нейромоделей

Для практических задач [12] проводились эксперименты по исследованию взаимосвязей между показателями качества нейромоделей на основе нейронных сетей: многослойной прямого распространения и радиально-базисной [2], а также нейро-нечетких сетей Мамдани и ANFIS [3]. Наиболее интересные графики экспериментально полученных зависимостей между показателями качества нейромоделей изображены на рис. 2–3.

Как видно из рис. 2, с увеличением L_{net} наблюдается рост I_{lr} , при возрастании $I_{div}(net, \langle x, y \rangle)$ отмечается убывание I_{gen} , с ростом I_{gen} наблюдается увеличение I_{ef} , с увеличением I_{apo} отмечается рост $I_{interp.}$, с ростом I_h наблюдается увеличение I_{eqf} и $I_{interp.}$

В свою очередь, из рис. 3 следует, что при возрастании I_{par} наблюдается увеличение I_{ef} , с ростом I_{hm} происходит убывание $I_{interp.}$, при возрастании I_{sym} наблюдается уменьшение I_{fb} и $I_{interp.}$, а также увеличение I_{hm} , при возрастании I_{pl} наблюдается увеличение I_{ef} .

Также в результате проведенных экспериментов отмечено, что с ростом сложности сети L_{net} уменьшается показатель соответствия нелинейностей выборки и нейромодели \tilde{I}_{nl} (при фиксированной выборке это объясняется увеличением нелинейности нейромодели), при возрастании показателя эмерджентности I_{α} наблюдается уменьшение разнообразия сети $I_{div}(net)$ и отношения разнообразия выборки и сети $I_{div}(net, \langle x, y \rangle)$, при возрастании $I_{div}(net)$ и $I_{div}(net, \langle x, y \rangle)$ наблюдается рост I_v , а также убывание показателей I_{ef} , I_{gen} , $I_{n.re}^{avg}$ и $I_{w.re}^{avg}$.

Увеличение I_{sym} и I_{sym}^u приводит к уменьшению I_{fb} и $I_{interp.}$, а также к увеличению I_{ha} , I_{hd} и I_{hm} . С ростом I_{ha} происходит увеличение I_{hm} . С ростом показателей I_{ha} , I_{hd} и I_{hm} происходит уменьшение показателей функциональной блочности I_{fb} и интерпретабельности $I_{interp.}$

С увеличением значения показателя автоматизации предорганизации в обучении I_{apo} наблюдается рост значений показателя интерпретабельности $I_{interp.}$, а также убывание показателей I_{ha} , I_{hd} , I_{hm} , I_{sym} и I_{sym}^u . Увеличение показателя иерархичности модели I_h приводит к возрастанию показателей эквивалентности I_{eqf} и интерпретабельности $I_{interp.}$

Увеличение показателя обобщения I_{gen} приводит к росту I_{ef} и $I_{n.re}^{avg}$. В свою очередь, увеличение значения $I_{n.re}^{avg}$ приводит к росту I_{ef} и $I_{w.re}^{avg}$, с возрастанием I_v происходит убывание $I_{n.re}^{avg}$, а увеличение I_{md} приводит к росту $I_{w.re}^{avg}$, $I_{n.re}^{avg}$, I_{ef} , I_{gen} и I_{par} , а также к убыванию $I_{div}(net)$ и $I_{div}(net, \langle x, y \rangle)$. При возрастании значений I_{par} наблюдается увеличение I_{ef} , $I_{n.re}^{avg}$ и $I_{w.re}^{avg}$, а также уменьшение $I_{div}(net)$, $I_{div}(net, \langle x, y \rangle)$ и I_v .

При увеличении значений N , N_n и N_w наблюдается уменьшение эмерджентности I_{α} . Возрастание N_w приводит к росту $I_{div}(net)$ и $I_{div}(net, \langle x, y \rangle)$, а также к убыванию I_{ef} , I_{gen} и $I_{n.re}^{avg}$. Увеличение I_{sp} приводит к увеличению I_{ef} , I_{md} , $I_{n.re}^{avg}$ и $I_{w.re}^{avg}$. С ростом значения I_{pl} возрастают I_{α} , I_{ef} , I_{md} , $I_{n.re}^{avg}$ и $I_{w.re}^{avg}$, с увеличением числа нейронов N_n происходит убывание I_{ef} , I_{pl} и I_{sp} , а также увеличение N_w , $I_{div}(net)$, $I_{div}(net, \langle x, y \rangle)$ и I_v .

Полученные качественные зависимости между показателями, характеризующими свойства нейромоделей, могут быть учтены при формировании интегральных показателей качества для исключения качественно эквивалентных дублирующихся показателей, что позволит упростить и ускорить вычисления.

ЗАКЛЮЧЕНИЕ

С целью автоматизации построения и анализа диагностических моделей в работе решена актуальная задача экспериментального исследования комплекса показателей качества диагностических нейромоделей.

Практическая ценность работы заключается в том, что при решении задач диагностики экспериментально исследован комплекс информационных показателей качества диагностических нейромоделей. Это позволило получить качественные зависимости между показателями, учет которых при формировании интегральных показателей качества позволяет существенно упростить вычисления путем исключения качественно эквивалентных (дублирующихся) показателей.

Работа выполнена в рамках госбюджетной темы кафедры программных средств Запорожского национального технического университета «Информационные технологии автоматизации распознавания образов и принятия решений для диагностики в условиях неопределенности на основе гибридных нечеткологических, нейросетевых и мультиагентных методов вычислительного интеллекта» (номер гос. регистрации 0109U007673).

СПИСОК ЛИТЕРАТУРЫ

1. Биргер, И. А. Техническая диагностика / И. А. Биргер. – М.: Машиностроение, 1978. – 240 с.
2. Хайкин, С. Нейронные сети: полный курс / С. Хайкин; пер. с англ. – М.: Вильямс, 2006. – 1104 с.
3. Субботин, С. О. Подання й обробка знань у системах штучного інтелекту та підтримки прийняття рішень: навчальний посібник / С. О. Субботин. – Запоріжжя: ЗНТУ, 2008. – 341 с.
4. Субботин, С. А. Методика и критерии сравнения моделей и алгоритмов синтеза искусственных нейронных сетей / С. А. Субботин // Радіоелектроніка, інформатика, управління. – 2003. – № 2. – С. 109–114.
5. Субботин, С. А. О сравнении нейросетевых моделей / С. А. Субботин // Нейроинформатика и ее приложения: материалы XI Всероссийского семинара, 3–5

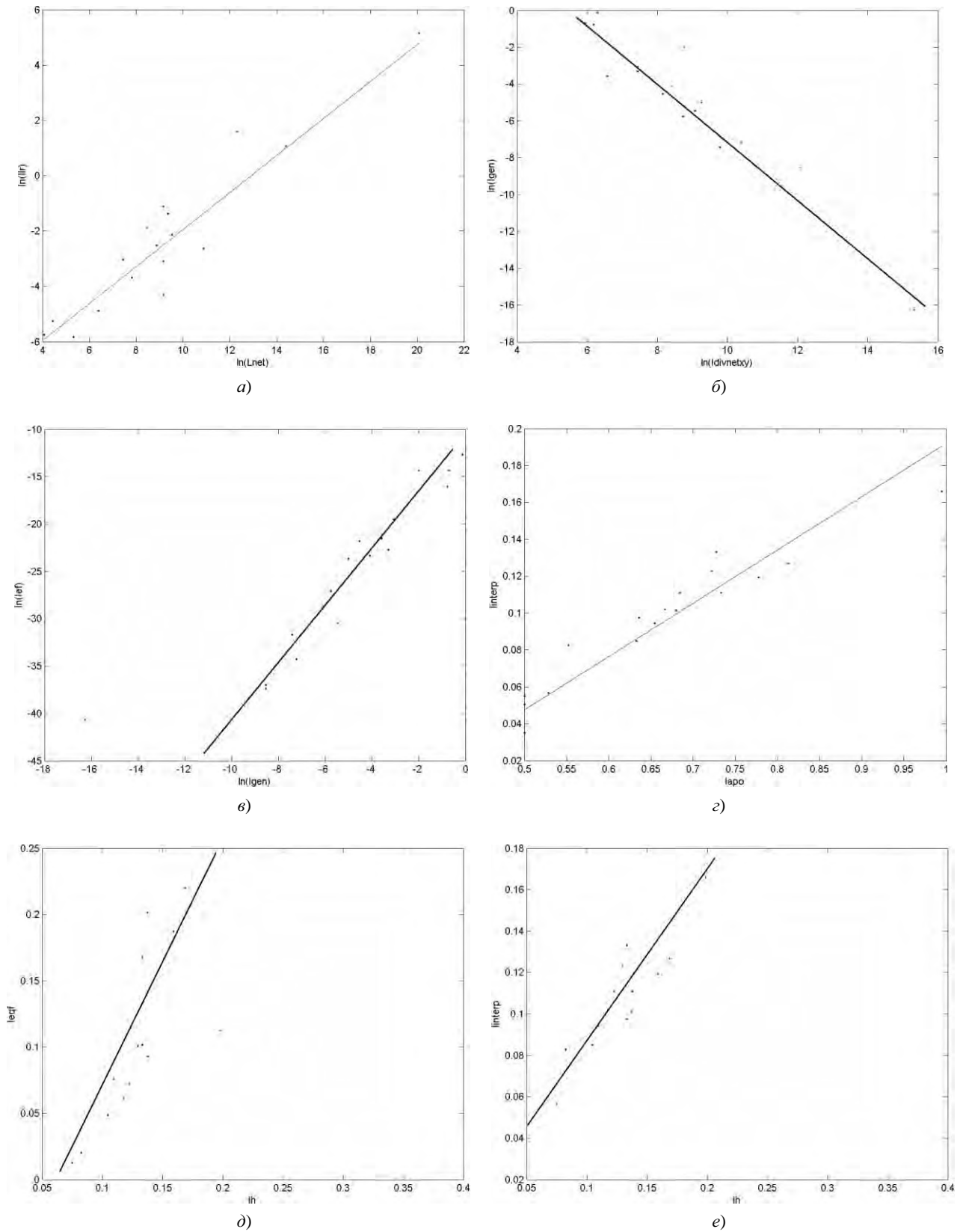


Рис. 2. Графики зависимостей:

a – $\ln(I_r)$ от $\ln(L_{net})$; *б* – $\ln(I_{gen})$ от $\ln(I_{div}(net, <x, y>))$; *в* – $\ln(I_{ef})$ от $\ln(I_{gen})$; *г* – I_{interp} от I_{apo} ; *д* – I_{ef} от I_h ; *е* – I_{interp} от I_h

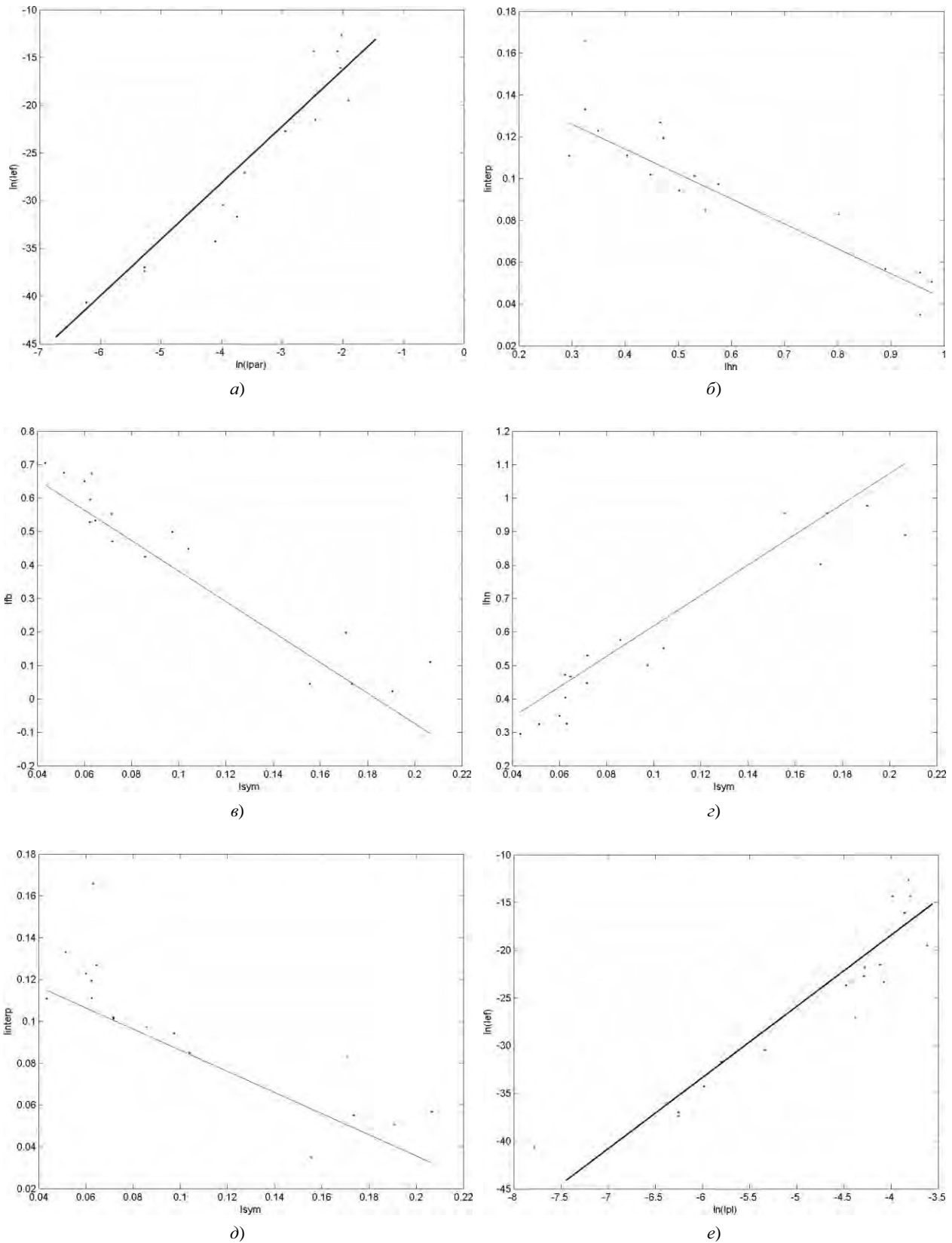


Рис. 3. Графики зависимостей:

$a - \ln(I_{ef})$ от $\ln(I_{par})$; $б - I_{interp}$ от I_{hn} ; $в - I_{fb}$ от I_{sym} ; $г - I_{hn}$ от I_{sym} ; $д - I_{interp}$ от I_{sym} ; $е - \ln(I_{ef})$ от $\ln(I_{pl})$

- октября 2003 г. / под ред. А. Н. Горбаня, Е. М. Мирке-са. – Красноярск : ИВМ СО РАН, 2003. – С. 152–153.
6. *Субботин, С. А.* Анализ свойств и критерии сравнения нейросетевых моделей для решения задач диагностики и распознавания образов / С. А. Субботин // Реестрация, зберігання і обробка даних. – 2009. – Т. 11. – № 3. – С. 42–52.
 7. *Субботин, С. А.* Модели критериев сравнения нейронных и нейро-нечетких сетей в задачах диагностики и классификации образов / С. А. Субботин // Наукові праці Донецького національного технічного університету. Серія «Інформатика, кібернетика та обчислювальна техніка». – Донецьк : ДНТУ, 2010. – Вип. 12 (165). – С.148–151.
 8. *Субботин, С. А.* Показатели качества и эффективность нейромоделей / С. А. Субботин // Проведение научных исследований в области обработки, хранения, передачи и защиты информации. Всероссийская конференция с элементами научной школы для молодежи, 1–5 декабря 2009 г., Ульяновск : сборник научных трудов. – Т. 2. – Ульяновск : УлГТУ, 2009. – С. 38–47.
 9. *Субботин, С. А.* Интегральные показатели качества диагностических нейромоделей / С. А. Субботин // Научная сессия МИФИ – 2010. XII Всероссийская научно-техническая конференция «Нейроинформатика-2010» : сборник научных трудов. – М. : МИФИ, 2010. – Ч. 1. – С. 12.
 10. *Субботин, С. А.* Комплекс характеристик и критериев сравнения обучающих выборок для решения задач диагностики и распознавания образов / С. А. Субботин // Математичні машини і системи. – 2010. – № 1. – С. 25–39.
 11. *Subbotin, S. A.* The Training Set Quality Measures for Neural Network Learning / S. A. Subbotin // Optical Memory and Neural Networks (Information Optics). – 2010. – Vol. 19. – № 2. – P. 126–139.
 12. *Богуслаев, А. В.* Прогрессивные технологии моделирования, оптимизации и интеллектуальной автоматизации этапов жизненного цикла авиационных дви-

гателей: монография / А. В. Богуслаев, Ал. А. Олейник, Ан. А. Олейник, Д. В. Павленко, С. А. Субботин ; под ред. Д. В. Павленко, С. А. Субботина. – Запорожье : Мотор Сич, 2009. – 468 с.

Надійшла 24.09.2010

Субботін С. О.

ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ Й АНАЛІЗ ВЗАЄМОЗВ'ЯЗКІВ ІНФОРМАЦІЙНИХ ПОКАЗНИКІВ ЯКОСТІ ДІАГНОСТИЧНИХ НЕЙРОМОДЕЛЕЙ

Експериментально досліджено комплекс інформаційних показників якості діагностичних нейромоделей при вирішенні практичних завдань. Отримано якісні залежності між показниками, урахування яких при формуванні інтегральних показників якості дозволяє істотно спростити обчислення шляхом виключення показників, що дублюються.

Ключові слова: якість, модель, технічна діагностика, нейронна мережа, нейро-нечітка мережа.

Subbotin S. A.

EXPERIMENTAL INVESTIGATION AND ANALYSIS OF INFORMATION QUALITY INDICES CORRELATION FOR DIAGNOSTIC NEUROMODELS

The complex of information quality indices of diagnostic neuromodels has been experimentally investigated by solving practical problems. Qualitative relationships between the indices have been obtained. When used for formation of integral quality indices, they can simplify computation by eliminating duplicate indices.

Key words: quality, model, technical diagnosis, neural network, neuro-fuzzy network.

ПРОГРЕСИВНІ ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

ПРОГРЕССИВНЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

PROGRESSIVE INFORMATION TECHNOLOGIES

УДК 004.03; 681.518:061

Евланов М. В.¹, Терещенко И. В.¹, Штангей С. В.¹

¹Канд. техн. наук, доцент Харьковского национального университета радиоэлектроники

РАЗРАБОТКА ИНФОРМАЦИОННОЙ ТЕХНОЛОГИИ МОНИТОРИНГА БИЗНЕС-ПРОЦЕССОВ ПРЕДПРИЯТИЯ

Статья посвящена разработке информационной технологии мониторинга бизнес-процессов предприятия, позволяющей автоматизировать процессы формирования представлений данного бизнес-процесса. Разработана структурная схема этой технологии и рассмотрено назначение основных ее элементов. Представлена методика настройки предложенной информационной технологии в зависимости от особенностей конкретного бизнес-процесса.

Ключевые слова: мониторинг, автоматизация, компонент, корректировка, онтологическая модель, синтез.

ОСНОВНЫЕ ЗАДАЧИ МОНИТОРИНГА БИЗНЕС-АКТИВНОСТИ ПРЕДПРИЯТИЯ

В настоящее время существует большая потребность в специализированных информационных технологиях (ИТ) оперативного мониторинга бизнес-активности (Business Activity Monitoring, BAM) предприятия. Концепция BAM была предложена аналитической компанией Gartner. В соответствии с этой концепцией BAM рассматривается как предоставление доступа в режиме, близком к режиму реального времени, к наиболее важным индикаторам производительности бизнеса с целью увеличения скорости и эффективности бизнес-операций [1]. Типичный сценарий BAM упрощенно выглядит следующим образом: лицо, принимающее бизнес-решения, оценивает происходящие на предприятии экономические события. При этом используется специализированное программное обеспечение, основанное на наборе

правил фильтрации событий и уведомления пользователей о возникающих событиях.

Основными задачами BAM являются [1]:

- измерение фактических параметров бизнес-процесса (БП) и контроль их отклонений от заданных параметров;
- контроль исполнения регламентов;
- обнаружение нестандартных ситуаций;
- персонализированный контроль за исполнителями БП;
- контроль нагрузки на сотрудников и загрузки оборудования.

BAM тесно связан с интеграцией корпоративных приложений. Для полноценного обеспечения мониторинга бизнес-событий в реальном времени на крупном предприятии необходимо интегрировать все используемые информационные службы.

Решение задач BAM обеспечивает возможность оперативно отслеживать и обрабатывать важнейшие

бизнес-события (новую сделку, изменение цепочки поставок, изменение законодательства, запрос на доработку ИТ и т. д.). Внедрение и эксплуатация систем ВАР способствует принятию быстрых, обоснованных и четких управленческих решений, результатами которых являются конкурентные преимущества, получаемые предприятием. Поэтому разработка специализированных ИТ ВАР является актуальной научно-прикладной задачей.

**ФОРМАЛИЗОВАННОЕ ОПИСАНИЕ
ИНФОРМАЦИОННОЙ ТЕХНОЛОГИИ
МОНИТОРИНГА БИЗНЕС-АКТИВНОСТИ
ПРЕДПРИЯТИЯ**

Для решения проблемы построения ИТ ВАР в работе [2] предложена обобщенная модель, описывающая взаимодействие основных компонентов данной ИТ. Эта модель имеет вид

$$L^{IT} = [L^{ISO}, L^{IM}, L^{AM}, L^{VM}, F_{IM}^{ISO}, F_{ISO}, F_{AM}^{ISO}, F_{AM}^{IM}, F_{ISO}^{VM}, F_{AM}^{VM}, F_{AM}^{ISO}, F_{AM}^{IM}, F_{AM}^{VM}], \quad (1)$$

где L^{IT} – категория структурированных множеств, описывающая ИТ мониторинга БП; L^{ISO} – подкатегория, описывающая изоморфные алфавиты языков описания БП; L^{IM} – структурно-категорная модель информационного представления БП; L^{AM} – структурно-категорная модель аналитического представления БП; L^{VM} – структурно-категорная модель визуального представления БП; F_{IM}^{ISO} – функтор, отображающий подкатеорию L^{ISO} в подкатеорию L^{IM} ; F_{ISO}^{IM} – функтор, отображающий подкатеорию L^{IM} в подкатеорию L^{ISO} ; F_{AM}^{ISO} – функтор, отображающий подкатеорию L^{ISO} в подкатеорию L^{AM} ; F_{AM}^{IM} – функтор, отображающий подкатеорию L^{AM} в подкатеорию L^{ISO} ; F_{VM}^{ISO} – функтор, отображающий подкатеорию L^{ISO} в подкатеорию L^{VM} ; F_{ISO}^{VM} – функтор, отображающий подкатеорию L^{VM} в подкатеорию L^{ISO} ; F_{AM}^{IM} – функтор, отображающий подкатеорию L^{IM} в подкатеорию L^{AM} и устанавливающий правила преобразования информационного представления БП в аналитическое представление; F_{VM}^{AM} – функтор, отображающий подкатеорию L^{AM} в подкатеорию L^{VM} и устанавливающий правила преобразования аналитического представления модели БП в визуальное представление.

ПОСТАНОВКА ЗАДАЧИ ИССЛЕДОВАНИЯ

Как следует из выражения (1), для детализации формализованного описания ИТ ВАР необходимо в первую очередь разработать формализованные описания информационного, визуального и аналитического представления БП. Результаты разработки

структурно-категорной модели, описывающей информационное представление БП, рассмотрены в [3]. Результаты разработки структурно-категорной модели, описывающей визуальное представление БП, рассмотрены в [4]. Результаты разработки структурно-категорной модели, описывающей информационное представление БП, рассмотрены в [5].

Согласно модели (1) преобразование формализованных описаний наблюдаемого БП может осуществляться двумя основными способами:

а) первый способ предполагает организацию преобразования формализованных описаний путем выполнения стандартных преобразований таких описаний в процессе эксплуатации ИТ (преобразования описаны функторами F_{AM}^{IM} и F_{VM}^{AM});

б) второй способ предполагает организацию взаимного преобразования формализованных описаний через использование единого механизма изоморфных преобразований (механизм описан подкатеорией L^{ISO} и функторами F_{IM}^{ISO} , F_{ISO}^{IM} , F_{AM}^{ISO} , F_{ISO}^{AM} , F_{VM}^{ISO} и F_{ISO}^{VM}).

Поскольку первый способ основан на стандартных программно-реализованных преобразованиях, в статье основное внимание уделено рассмотрению практических подходов к реализации второго способа в рамках ИТ мониторинга БП предприятия.

**РАЗРАБОТКА СТРУКТУРНОЙ СХЕМЫ
ИНФОРМАЦИОННОЙ ТЕХНОЛОГИИ
МОНИТОРИНГА БИЗНЕС-ПРОЦЕССОВ
ПРЕДПРИЯТИЯ**

В настоящее время структура типовой ИТ мониторинга ВАР имеет вид, показанный на рис. 1 [6].

В качестве основных задач, которые должна решать типовая ИТ ВАР, выделяются такие задачи [7]:

– обеспечить возможность мгновенного осознания ситуации, то есть оценки того, как повлияют события (неисправности в сети, состояние баз данных,



Рис. 1. Структурная схема типовой информационной технологии мониторинга бизнес-деятельности предприятия

изменения в Web, изменение показателей каких-то ресурсов) на ход операций в бизнесе;

- осознав ситуацию, принимать решения и передавать управляющие воздействия, которые могут изменить бизнес-процессы;

- фиксировать нарушения или попытки нарушения политики и соглашений об уровне обслуживания;

- накапливать статистику производительности всех операций.

Такой подход к построению ИТ мониторинга БП предприятия определяет следующие основные блоки этой технологии:

а) хранилище данных (ХД) реального времени, в котором хранятся описания событий, происходящих в аппаратном обеспечении и программных приложениях, а также другие данные, непосредственно связанные с течением бизнес-процессов предприятия;

б) сервер ВАР, который управляет ХД реального времени в соответствии с разработанными моделями процессов, а также управляет пересылкой данных из ХД в кэш-память и доступностью этих данных для аналитических инструментов и средств подготовки отчетов;

в) оперативные процессы, под которыми понимаются функциональные задачи и модули ИСУП, в частности, систем ERP, SCM, CRM, Human-Resources Management и им подобных, являющиеся источниками описаний событий, происходящих в аппаратном обеспечении и программных приложениях, а также других данных, непосредственно связанные с течением бизнес-процессов предприятия;

г) аналитическая машина, которая на основе данных из ХД рассчитывает значения ключевых показателей производительности (Key Performance Indicators, KPI), определяющих степень достижения поставленных перед БП предприятия целей, а также осуществляет ретроспективный анализ БП и прогнозирование бизнес-деятельности предприятия;

д) машина для правил, которая осуществляет выявление новых и модификацию существующих бизнес-правил выполнения БП предприятия на основе информации, поступающей из аналитической машины;

е) пульт управления, который представляет собой набор способов и средств визуализации результатов работы аналитической машины и машины для правил. Предназначен для отображения текущего состояния БП, результатов анализа и прогноза развития БП, а также сигналов о возникающих бизнес-событиях и чрезвычайных ситуациях.

Рассмотренная схема ИТ мониторинга БП предприятия основана на использовании следующих ос-

новных способов формализованного представления БП:

- визуальное представление, используемое в процессе функционирования пульта управления для визуализации результатов функционирования аналитической машины и машины для правил;

- аналитическое представление, используемое в процессе функционирования аналитической машины и, в ряде случаев, машины для правил;

- представление на уровне хранимых данных или информационное представление, используемое в процессе функционирования ХД реального времени и сервера ВАР.

Для обеспечения возможности автоматизированного формирования этих представлений БП необходимо модифицировать схему, приведенную на рис. 1. С учетом результатов разработки формализованных описаний БП [3–5] и категорной модели, определяющей возможности автоматизированного преобразования этих представлений [2], структурная схема ИТ мониторинга БП предприятия примет вид, показанный на рис. 2.

Назначение таких компонентов разрабатываемой ИТ мониторинга БП предприятия, как ХД мониторинга БП, сервер мониторинга БП, аналитическая машина, машина правил и пульт управления, совпадает с назначением аналогичных компонентов типовой ИТ ВАР, структура которой показана на рис. 1.

Компонент разрабатываемой ИТ «CASE-модуль синтеза схемы ХД» предназначен для визуального проектирования и корректировки схемы ХД или витрины данных, являющейся формализованным описанием информационного представления наблюдаемого БП.

Компонент разрабатываемой ИТ «CASE-модуль синтеза дерева целей и системы сбалансированных показателей (ССП)» предназначен для визуального проектирования и корректировки дерева целей и соответствующего каждой цели подмножества сбалансированных показателей достижимости этой цели. Результаты разработки дерева целей и связанной с ним SSP являются формализованным описанием аналитического представления наблюдаемого БП.

Компонент разрабатываемой ИТ «Модуль разработки/импорта визуальных моделей БП» предназначен для выполнения следующих функций:

- проектирования визуальных моделей наблюдаемого БП в одной из трех нотаций визуального моделирования БП (IDEF0, Use Case, Activity);

- импорта визуальных моделей наблюдаемого БП, которые разработаны по результатам предварительного обследования объекта автоматизации и его БП

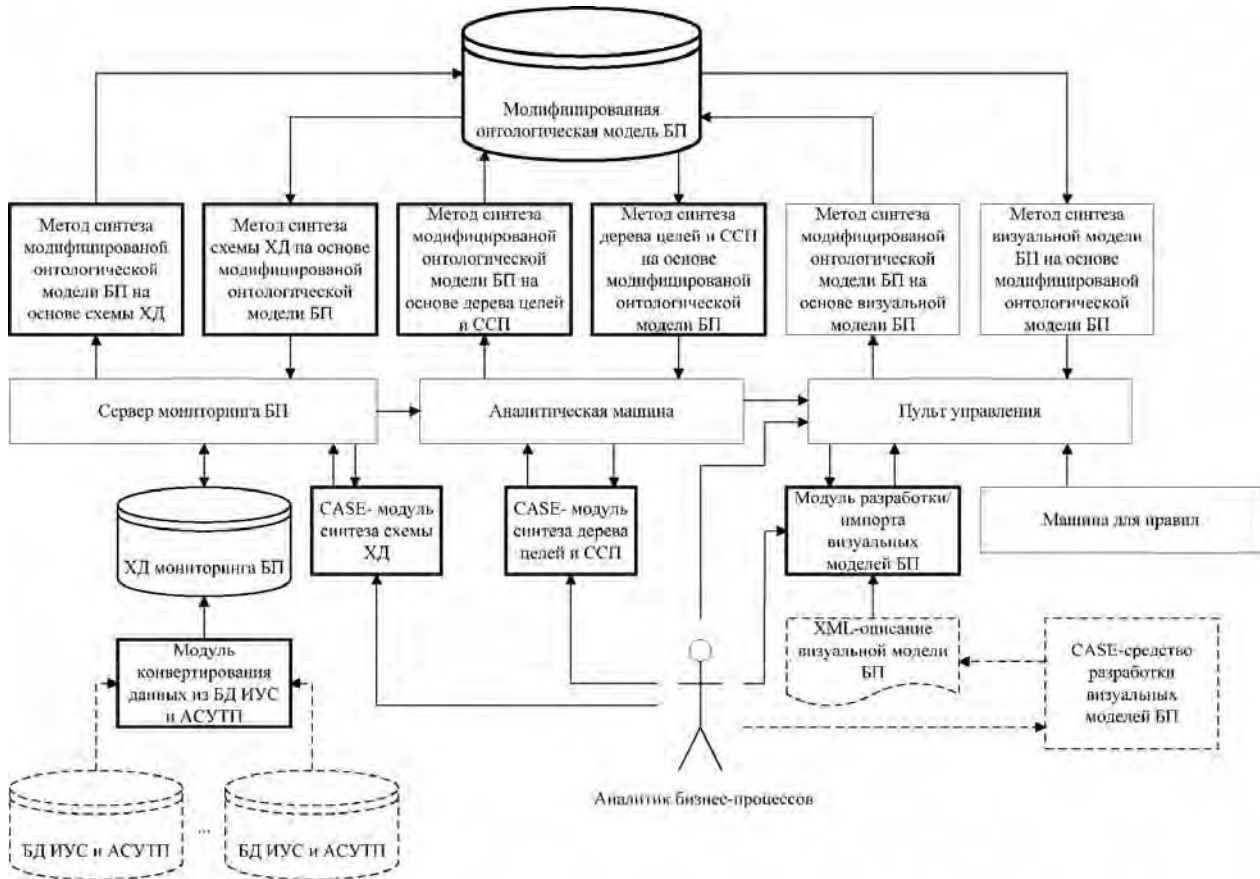


Рис. 2. Схема взаимодействия основных структурных компонентов информационной технологии мониторинга бизнес-процессов предприятия

с использованием традиционных CASE-средств (All-Fusion Modelling Suite, Rational Rose и т. п.), показанных на рис. 2 пунктиром.

Подобный импорт производится в виде XML-документов (показан на рис. 2 пунктиром), описывающих разработанную визуальную модель БП. Структура такого описания является стандартной и определяется онтологической моделью БП.

Компонент разрабатываемой ИТ «Модифицированная онтологическая модель БП» представляет собой шину метаданных [8], описывающих наблюдаемый БП как совокупность согласованных между собой атрибутов. Эти атрибуты характеризуют наблюдаемый БП с точки зрения его информационного, аналитического и визуального представления [9]. Формирование значений этих атрибутов происходит на основе результатов работы компонентов «CASE-модуль синтеза схемы ХД», «CASE-модуль синтеза дерева целей и ССП» и «Модуль разработки/импорта визуальных моделей БП», а сформированные значения могут использоваться для корректировки существующих или синтеза новых представлений наблюдаемого БП в указанных компонентах.

Компоненты разрабатываемой ИТ «Метод синтеза модифицированной онтологической модели БП на основе схемы ХД», «Метод синтеза модифицированной онтологической модели БП на основе дерева целей и ССП», «Метод синтеза модифицированной онтологической модели БП на основе визуальной модели БП» представляют собой совокупность SQL-запросов и программно реализованных процедур обработки данных, обеспечивающих трансформацию описаний схемы ХД или витрину данных, дерева целей и ССП, или же визуальных моделей БП в набор значений атрибутов шины метаданных, описывающих этот БП.

Компоненты разрабатываемой ИТ «Метод синтеза схемы ХД на основе модифицированной онтологической модели БП», «Метод синтеза дерева целей и ССП на основе модифицированной онтологической модели БП», «Метод синтеза визуальной модели БП на основе модифицированной онтологической модели БП» представляют собой совокупность SQL-запросов и программно реализованных процедур обработки данных, обеспечивающих трансформацию совокупности значений атрибутов шины метаданных,

описывающих наблюдаемый БП в описания схемы ХД или ВД, дерева целей и ССП, или же визуальных моделей этого БП.

Компонент разрабатываемой ИТ «Модуль конвертирования данных из БД ИУС и АСУТП» предназначен для реализации автоматизированной процедуры выборки оперативных данных о текущем состоянии наблюдаемого БП из БД ИС и АСУТП, эксплуатируемых на предприятии, преобразования этих данных к требуемому формату и записи результатов преобразования в ХД или витрину данных наблюдаемого БП.

Рассмотренные структурные особенности предлагаемой ИТ определяют методику настройки этой ИТ на особенности конкретного автоматизированного БП. Данная методика состоит из следующих этапов.

Этап 1. Разработка аналитиком визуальной модели БП, чей мониторинг планируется осуществлять. Если разработка выполнялась с использованием модуля разработки/импорта визуальных моделей БП ИТ мониторинга автоматизированных БП (пункт меню «Моделирование», опция «Визуальная модель»), то перейти к Этапу 3. Если разработка выполнялась

с использованием традиционных CASE-средств, то импортировать синтезированную визуальную модель в XML-документ и перейти к Этапу 2.

Этап 2. Импорт XML-документа, содержащего описание визуальной модели БП, в программный модуль ИТ мониторинга БП предприятия.

Этап 3. Корректировка и сохранение разработанной или импортированной визуальной модели автоматизированного БП как его визуального представления. Результат сохранения импортированной визуальной модели БП показан на рис. 3.

Этап 4. Автоматический синтез модифицированной онтологической модели автоматизированного БП на основе его визуального представления путем заполнения схемы шины метаданных ИТ данными, полученными в результате выполнения Этапа 3.

Этап 5. Автоматический синтез и визуализация дерева целей автоматизированного БП на основе результатов выполнения Этапа 4. Пример визуализации синтезированного дерева целей БП удаления окалина металлопроката показан на рис. 4.

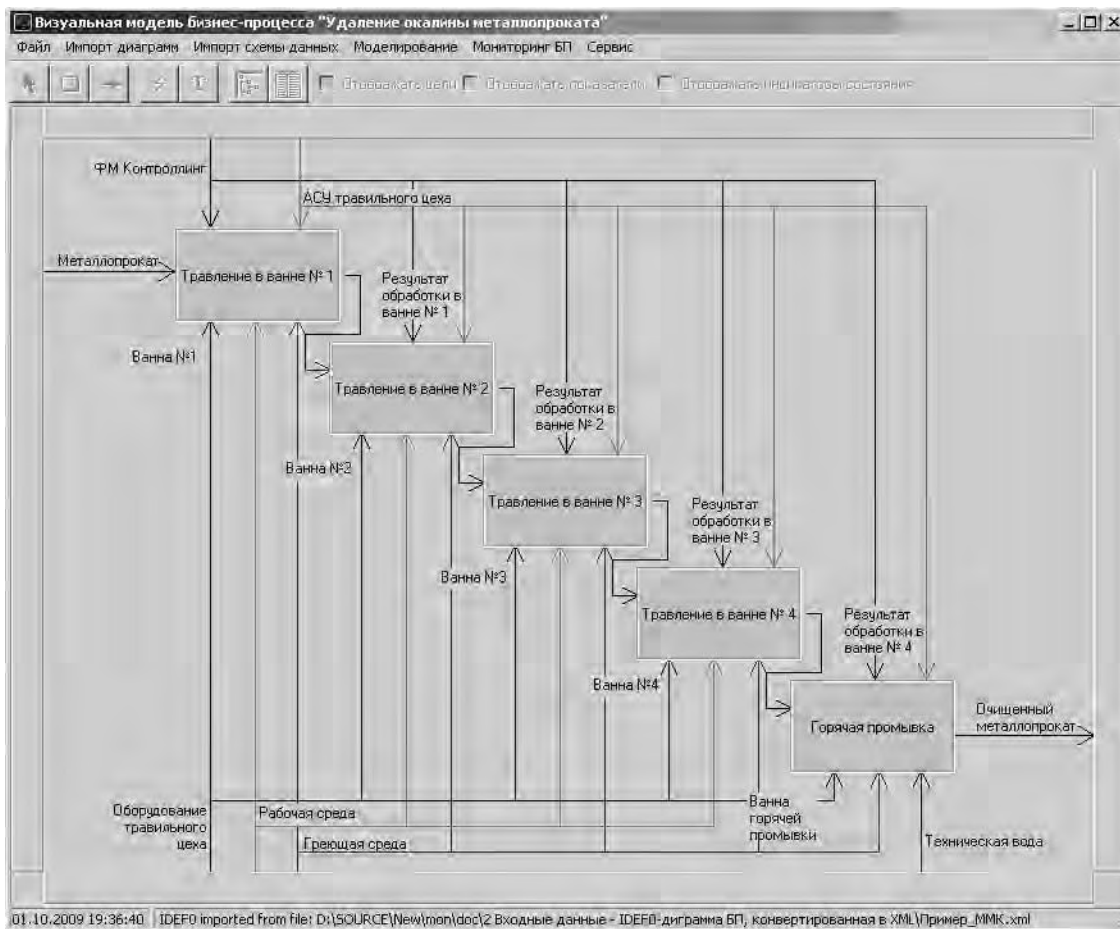


Рис. 3. Сохраненная IDEF0-модель как визуальное представление автоматизированного бизнес-процесса

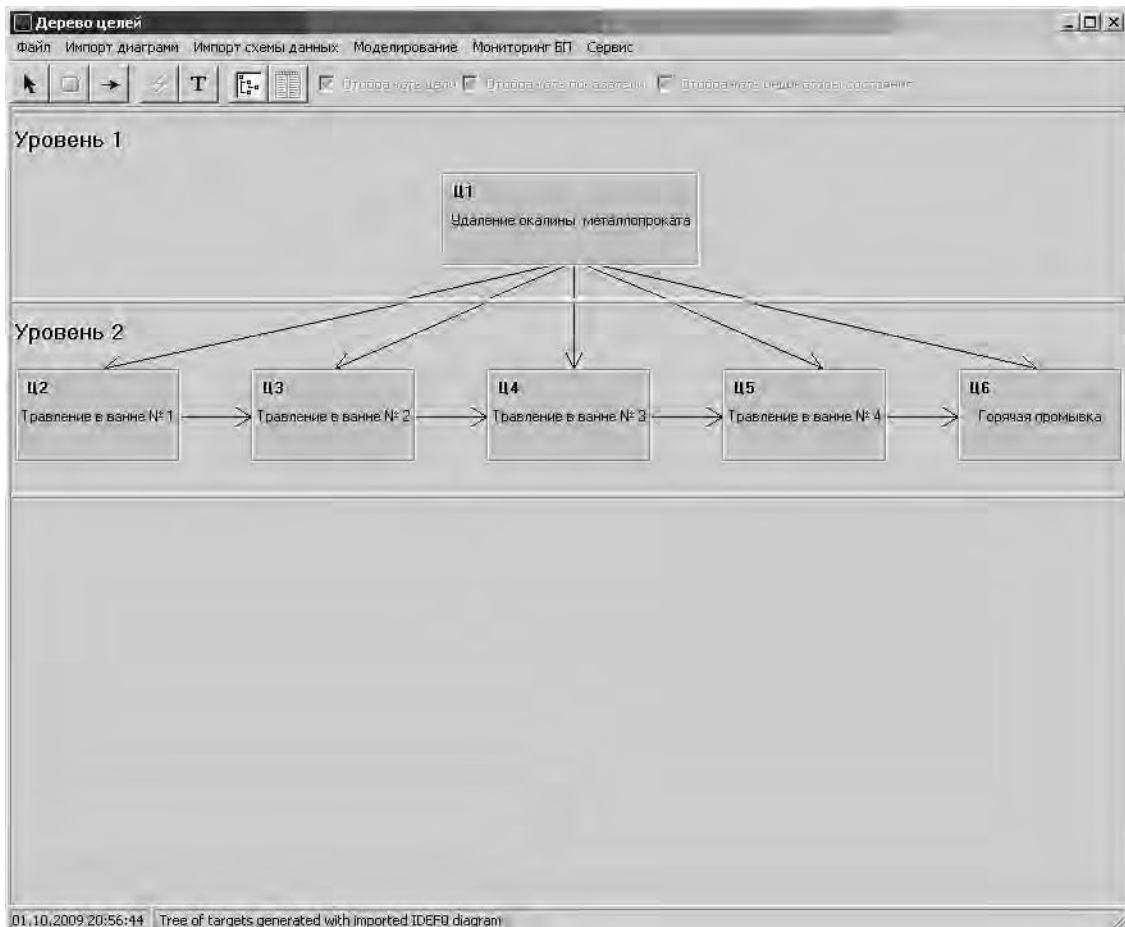


Рис. 4. Пример визуализации дерева целей бизнес-процесса, синтезированного на основе модифицированной онтологической модели этого бизнес-процесса

fmPokazatel

Система сбалансированных показателей для цели "Удаление окислы в ванне №1"

Настройка весовых коэффициентов: Весовые коэффициенты заданы корректно. Добавить показатель

Обобщенный показатель: Рассчитанное значение: 0,88. Дата расчета: 01.11.2009 21:06:09. Пересчитать

Наименование показателя	Текущее значение
Затраты на кислоту	0,8
Затраты на техническую воду	0,83
Затраты на подогрев рабочей среды	0,9
Затраты на з/п персонала травильного цеха	1,00

Общие сведения:

П1 (Название): Затраты на кислоту

В1 (Вес. коэф-т): 0,25

Тип: Затраты на рабочую среду

Min значение: 3,84. Ед. измерения: м3

Max значение: 5,76

Ц1 (Цена единицы): 10 000,00 грн.

Данные для расчета значения П1:

Таблица ВД: indicators

Атрибут: value

Агрегация: avg

Анализируемый период: 4,00 Часы

Применить. Удалить показатель

Рис. 5. Пример синтеза сбалансированного показателя достижимости цели наблюдаемого бизнес-процесса

Етап 6. Формирование дерева целей автоматизированного БП на основе результатов выполнения Этапа 4.

Етап 7. Корректировка и сохранение дерева целей автоматизированного БП, сформированного в результате выполнения Этапа 6.

Етап 8. Импорт схем данных БД ИС и АСУТП, которые являются источниками оперативных данных о текущем состоянии наблюдаемого автоматизированного БП.

Етап 9. Синтез перечня сбалансированных показателей достижимости каждой из целей, выделенных на Этапе 7, и формирование информационной модели каждого из показателей на основе результатов выполнения Этапа 8. В результате выполнения этого этапа для каждого из показателей устанавливается перечень атрибутов таблиц исходных БД и правило агрегации, согласно которому вычисляется значение этого показателя. Пример синтеза сбалансированного показателя показан на рис. 5.

Етап 10. Корректировка и сохранение ССП, синтезированной в результате выполнения Этапа 9.

Етап 11. Автоматическая корректировка модифицированной онтологической модели автоматизированного БП по результатам синтеза его аналитического представления путем дополнения и уточнения содержимого шины метаданных ИТ данными, полученными в результате выполнения Этапа 10.

Етап 12. Автоматический синтез и визуализация схемы ВД автоматизированного БП как его информационного представления на основе результатов выполнения Этапа 11.

Етап 13. Корректировка и сохранение схемы ВД, сформированной в результате выполнения Этапа 12.

Етап 14. Автоматическая корректировка модифицированной онтологической модели автоматизированного БП по результатам синтеза его информационного представления путем дополнения и уточнения содержимого шины метаданных ИТ данными, полученными в результате выполнения Этапа 13.

Етап 15. Настройка модуля конвертирования данных из БД ИУС и АСУТП на процесс конвертации данных из фрагментов БД ИСУП и АСУТП, выделенных как источники оперативных данных о текущем состоянии наблюдаемого автоматизированного БП.

ВЫВОДЫ

Предлагаемая структурная схема разрабатываемой ИТ мониторинга автоматизированных БП позволяет интегрировать данную ИТ в существующую совокупность средств информатизации управления предприятием без проведения реинжиниринга последних. Это позволяет сократить затраты времени и

финансов на подготовку объекта автоматизации к вводу разрабатываемой ИТ в действие.

Выполнение работ по настройке предложенной ИТ мониторинга БП предприятия в соответствии с данной методикой позволяет сократить время ввода в эксплуатацию предложенной ИТ с четырех-пяти месяцев до одной-двух недель.

СПИСОК ЛИТЕРАТУРЫ

1. Черняк, Л. На пути к предприятию, управляемому в реальном времени / Л. Черняк // Открытые системы. – 2002. – № 12. – С. 43–47.
2. Євланов, М. В. Формалізований опис технології візуального моделювання автоматизованих бізнес-процесів промислового підприємства / М. В. Євланов, В. О. Антонов, Є. В. Корнеєва // Вісник Академії митної служби України. – 2007. – № 3 (35). – С. 95–100.
3. Євланов, М. В. Формалізоване описання інформаційної моделі бізнес-процесу / М. В. Євланов, Є. В. Корнеєва // Системи обробки інформації. – 2009. – Вип. 6(80). – С. 167–171.
4. Антонов, В. А. Взаимное преобразование визуальных моделей бизнес-процессов промышленного предприятия / В. А. Антонов, Е. В. Корнеєва // Управління розвитком. Міжнародна науково-практична конференція «Стратегії ІТ-технологій в освіті, економіці та екології» 15–16 листопада 2007 р. – Харків : ХНЕУ, 2007. – № 7. – С. 20–21.
5. Євланов, М. В. Применение системы сбалансированных показателей в информационной технологии мониторинга автоматизированных бизнес-процессов / М. В. Євланов, В. А. Антонов, Е. В. Корнеєва // Системи обробки інформації. – 2010. – Вип. 5(86). – С. 211–216.
6. Черняк, Л. Мониторинг бизнес-процессов [Электронный ресурс] / Л. Черняк // Сайт журнала «Открытые системы». – Режим доступа: http://www.osp.ru/os/2005/10/380436/_p1.html, вільний. – Назва с экрана.
7. Черняк, Л. Сложные события и мониторинг бизнеса / Л. Черняк // Сайт журнала «Открытые системы». – Режим доступа: http://www.osp.ru/os/2005/02/185306/_p1.html, вільний. – Назва с экрана.
8. Корнеєва, Е. В. Интеграция моделей автоматизированного бизнес-процесса на основе шины метаданных / Е. В. Корнеєва // 14-й Международный молодежный форум «Радиоэлектроника и молодежь в XXI веке»: сб. материалов форума. Ч. 2. – Харьков : ХНУРЭ, 2010. – С. 125.
9. Корнеєва, Е. В. Автоматизация бизнес-процессов с помощью информационных технологий / Е. В. Корнеєва // Матеріали електронної техніки та сучасні інформаційні технології. Тези доповідей на Четвертій міжнародній науково-практичній конференції МЕТІТ-4: 19–21 травня 2010 р., Кременчук, Україна: збірник тез доповідей. – Кременчук: Кременчугський університет економіки, інформаційних технологій і управління, 2010. – С. 237–238.

Надійшла 20.10.2010

Євланов М. В., Терещенко І. В., Штангей С. В.
РОЗРОБКА ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ МОНІТОРИНГУ БІЗНЕС-ПРОЦЕСІВ ПІДПРИЄМСТВА

Стаття присвячена розробці інформаційної технології моніторингу бізнес-процесів підприємства, які дозволяють автоматизувати процеси формування представлень даного бізнес-процесу. Розроблено структурну схему цієї технології і розглянуто призначення основних її елементів. Представлено методику настройки запропонованої інформаційної технології у залежності від особливостей конкретного бізнес-процесу.

Ключові слова: моніторинг, автоматизація, компонент, коригування, онтологічна модель, синтез.

Evlanov M. V., Terecsenko I. V., Shtangey S. V.

DEVELOPMENT OF INFORMATION TECHNOLOGY OF ENTERPRISE BUSINESS PROCESSES MONITORING

The paper describes the information technology of enterprise business processes monitoring enabling automatic formation of given business process representations. The block diagram

of this technology is developed and the function of its basic elements is considered. The technique of information technology adjustment is presented taking into account the features of the specified business process.

Key words: monitoring, automation, component, adjustment, ontological model, synthesis.

УДК 681.324

Зинченко Ю. Е.¹, Гриценко А. А.¹, Зеленева И. Я.¹, Войтов Г. В.²

¹Канд. техн. наук, доцент Донецького національного технічного університета

²Ассистент Донецького національного технічного університета

ПРОГРАММНО-АППАРАТНАЯ БИБЛИОТЕКА МАТЕМАТИЧЕСКИХ ФУНКЦИЙ ДЛЯ СИСТЕМ НА ПЛИС

В статье предлагается метод создания библиотеки математических функций для систем на ПЛИС (SoPC) и набора необходимых для работы с ней аппаратно-программных средств. Предлагаемая библиотека дает возможность варьировать структуру системы на ПЛИС в соответствии с требованиями пользователя и доступными аппаратными ресурсами.

Ключевые слова: ПЛИС, система на ПЛИС, процессор, периферийный модуль, библиотека.

ВВЕДЕНИЕ

В настоящее время производители ПЛИС предоставляют широкие возможности по созданию систем на ПЛИС с использованием различных микропроцессорных архитектур. В частности, речь идет об архитектурах Xilinx MicroBlaze [1] и Altera Nios II [2], которые подразумевают наличие центрального микропроцессора и набора периферийных модулей, взаимодействующих посредством предоставляемого процессором шинного интерфейса.

Программное обеспечение рассматриваемых систем создается с использованием современных высокоуровневых языков программирования, в частности С и С++ [3, 4]. Оба эти языка предоставляют доступ

к стандартной библиотеке математических функций. Необходимость использования функций математической библиотеки ведет к увеличению вычислительной нагрузки на центральный процессор.

ПОСТАНОВКА ЗАДАЧИ

Рассматриваемые в данной работе системы на ПЛИС характеризуются наличием специфичных модулей, которые обеспечивают решение узкоспециализированных задач, позволяя центральному процессору переложить на них обязанности в области трудоемких вычислений (рис. 1), в частности в области вычисления математических функций.

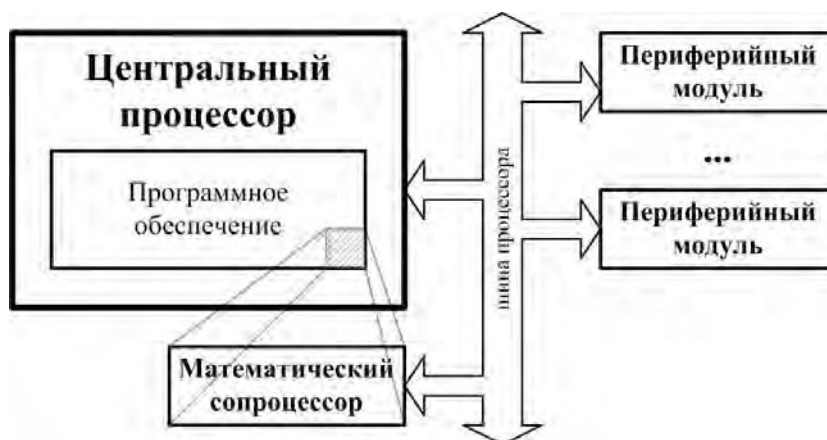


Рис. 1. Архитектура микропроцессорной системы на ПЛИС, включающей сопроцессор

© Зинченко Ю. Е., Гриценко А. А., Зеленева И. Я., Войтов Г. В., 2011

Важным вопросом, который возникает в процессе создания таких программно-аппаратных систем, является организация взаимодействия сопроцессоров и центрального процессора. С точки зрения архитектуры это подразумевает создание дополнительного программного модуля, обеспечивающего доступ к каждому из сопроцессоров, а также реализацию аппаратных модулей, используемых для подключения сопроцессоров к шине процессора. Для решения этого вопроса в данной работе предлагается подход, который позволяет изолировать логику математического сопроцессора от деталей его подключения к шине процессора, а логику программного обеспечения – от деталей взаимодействия с сопроцессором.

Использование аппаратного модуля для выполнения вычислений является, с одной стороны, производительным вариантом оптимизации, а с другой – ресурсоемким. Поэтому реализация таких модулей требует использования подхода, который позволяет получить лучшие значения обеих характеристик в конкретном случае. Описание соответствующего подхода является вторым важным вопросом, который рассматривается в данной работе. Предлагаемое решение этого вопроса базируется на организации диалога с экспертом, обладающим знаниями предметной области реализуемого системой приложения. Целью такого диалога является выявление ограничений использования тех вычислительных компонент системы, которые будут реализованы в форме математических сопроцессоров.

ИСПОЛЬЗОВАНИЕ ПРОГРАММНО-АППАРАТНОЙ БИБЛИОТЕКИ И АРХИТЕКТУРА ПОДКЛЮЧЕНИЯ СОПРОЦЕССОРОВ

Согласно [5], в процессе разработки программно-аппаратной системы пользователь не должен самостоятельно определять способ реализации отдельных компонент. Для обеспечения работы пользователя требуется, во-первых, иметь некоторую формальную методику разработки систем на ПЛИС, обеспечивающую автоматизированный анализ ее структуры и поиск путей ее оптимизации. Во-вторых, обеспечить пользователя набором средств, необходимых для того, чтобы, с одной стороны, позволить ему управлять процессами формирования структуры системы, с другой – обеспечить качественную его изоляцию от деталей выполнения этих процессов.

Исходными данными будем считать некоторую систему на ПЛИС, использующую функции математической библиотеки. Здесь следует отметить необходимость того, чтобы разработчик использовал не

стандартную математическую библиотеку, а специальный ее аналог, обеспечивающий дополнительную функциональность, которая необходима для дальнейшего анализа. В частности, речь идет о поддержке механизмов профилирования.

Первым этапом является предварительный анализ, который представлен на рис. 2 с использованием нотации UML [6]. На этом этапе выполняется получение результатов профилирования программной части. Эти результаты генерируются предоставляемым, специфичным для конкретной архитектуры, вариантом математической библиотеки.

Анализ результатов показывает основные возможности оптимизации системы. Дальнейшие этапы могут повторяться циклически, для получения наиболее подходящего варианта относительно повышения производительности и расхода аппаратных ресурсов.

Вторым этапом является получение экспертных знаний о том, каким образом должны или могут выполняться вычисления математических функций, которые были отобраны на первом этапе. Точность и полнота экспертных знаний влияет на результаты оптимизации системы. В качестве таких знаний могут выступать, например:

– *Способ представления чисел.* Если необходимо использовать представление с плавающей запятой, то, скорее всего, нужно будет использовать сторонние ядра для выполнения арифметических операций; если можно использовать представление с фик-

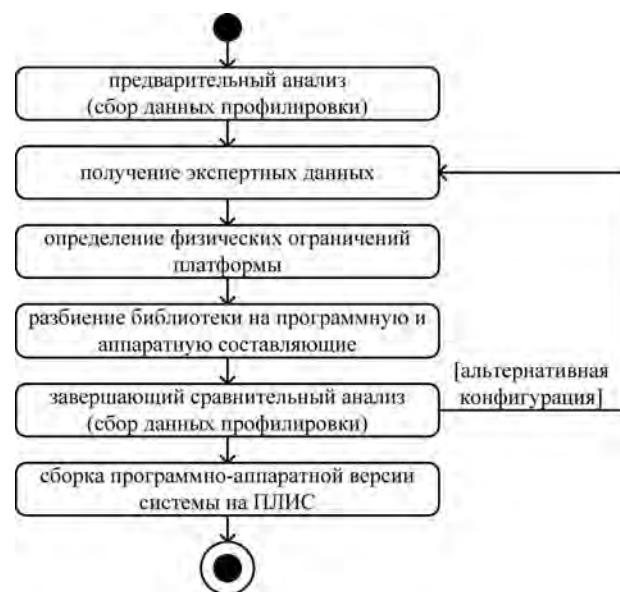


Рис. 2. Методика разработки системы на ПЛИС, использующей программно-аппаратную реализацию математической библиотеки



Рис. 3. Архитектура для подключения математического сопроцессора

сированной запятой, то понадобятся соответствующие функции преобразования.

– *Необходимая точность результатов.* Допустим известно то, что требуемая точность не превышает некоторое количество разрядов после запятой. В этом случае можно сократить количество ресурсов, которые необходимы для размещения сопроцессора.

– *Допустимый диапазон входных значений.* Программная реализация часто отличается общностью, с другой стороны, аппаратная реализация позволяет повысить производительность и уменьшить затраты ресурсов за счет своей специфичности.

Увеличение объема информации дает возможность создавать более специфичные и, как следствие, более производительные и менее затратные по ресурсам сопроцессоры.

Следующим шагом является получение информации об используемой платформе. Эта информация позволяет выбрать наиболее подходящую для конкретной платформы реализацию аппаратного модуля, а также определить способ подключения модуля к шине процессора. Также появляется возможность определить механизмы доступа к модулю со стороны программного обеспечения.

Например, если известно, что будет использоваться ПЛИС FPGA Altera Cyclone II, то можно выбрать реализацию модуля, оптимизированную для архитектуры этой ПЛИС [7]. С другой стороны, подключение модуля к микропроцессору будет осуществляться через шину Avalon [8], что позволяет выбрать необходимый модуль сопряжения с шиной. Кроме того, можно однозначно определить способ взаимодействия с модулем со стороны программного обеспечения [3].

Предлагается использовать архитектуру подключения сопроцессоров, приведенную на рис. 3.

В предлагаемой архитектуре выделен модуль подключения. Этот модуль позволяет использовать один

неоптимизированный вариант сопроцессора для нескольких платформ. Он должен обеспечивать корректный протокол работы с сопроцессором. Например, в случае использования интерфейса отображения в памяти шины Avalon [8] модуль обеспечивает запись и чтение данных по соответствующим адресам.

Следующим этапом разработки является разбиение математической библиотеки на программную и аппаратную части. На этом этапе возникает необходимость в получении от пользователя информации о том, сколько и каких ресурсов ПЛИС можно использовать. К этому моменту точно известно, сколько ресурсов необходимо для размещения того или иного сопроцессора, поэтому, в зависимости от имеющихся ресурсов, можно определить все допустимые конфигурации системы.

Последним этапом является сравнительный анализ. На этом этапе используется профилирующая версия математической библиотеки, что позволяет оценить полученный в результате оптимизации прирост производительности. По необходимости разработчик получает возможность сравнить несколько альтернативных конфигураций системы.

СТРУКТУРА ПРОГРАММНО-АППАРАТНОЙ БИБЛИОТЕКИ

Следует отметить, что с точки зрения системы на ПЛИС, структура библиотеки в процессе разработки изменяется. При использовании языков C и C++ библиотека будет состоять изначально из двух, а позже из трех файлов. Заголовочный файл обеспечивает возможность взаимодействия с библиотекой, а два файла исходных кодов (или два объектных файла) содержат реализацию программной составляющей и модулей доступа к сопроцессорам (рис. 3). Аппаратная часть библиотеки зависит от того, какие функции и в какой конфигурации погружены в аппаратуру.



Рис. 4. Первоначальная структура библиотеки

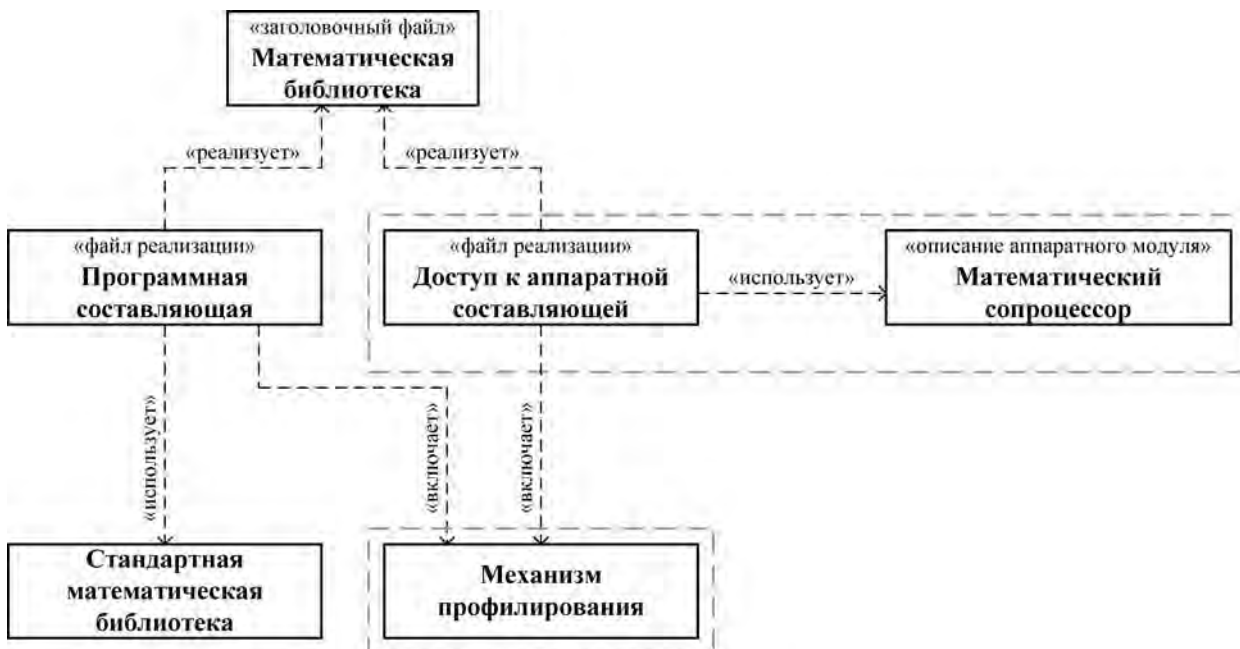


Рис. 5. Окончательный вариант структуры библиотеки

В процессе разработки начальной версии системы и на этапе предварительного анализа программно-аппаратная библиотека является оберткой стандартной библиотеки и имеет полностью программную реализацию. Ее основным отличием является то, что она позволяет осуществить получение результатов профилирования (рис. 4).

По завершении сбора экспертной информации и данных об используемой платформе выполняется генерация специализированной версии библиотеки, которая все же позволяет осуществить профилирование (рис. 5).

Это нужно для выполнения сравнительного анализа исходной и оптимизированной версий, а также, в

случае необходимости, нескольких разных оптимизированных версий.

Механизмы профилирования удаляются из библиотеки только после того, как будет выбрана подходящая пользователю конфигурация.

ПРИМЕР РЕАЛИЗАЦИИ ЭКСПОНЕНЦИАЛЬНОЙ ФУНКЦИИ

Допустим, что согласно результатам профилирования программного обеспечения для системы на ПЛИС значительная часть вычислительных ресурсов расходуется на вычисление экспоненциальной функции [9]. Согласно ранее предложенной методике, следующим этапом является получение ограниче-

```

double hs_math_exp (double x) {
    assert (0.0 <= x && x <= 2.0);
    Xuint32 x_fixed = to_fixed (x);
    }
    XIo_Out32 (HS_MATH_EXP_NPU, x_fixed);
    Xuint32 y_fixed = XIo_In32 (HS_MATH_EXP_NPU);
    }
    double y = from_fixed (y_fixed);
    assert (1.0 <= y && y <= 7.38906);
    assert (are_equal (y, exp (x), 5));
    return y;
}
    
```

Шаг 1. Контроль и подготовка исходных данных

Шаг 2. Расчет с использованием сопроцессора

Шаг 3. Контроль, подготовка и выдача результатов

Рис. 6. Пример реализации специализированного модуля доступа к сопроцессору, вычисляющему экспоненциальную функцию (для архитектуры Xilinx MicroBlaze [1])

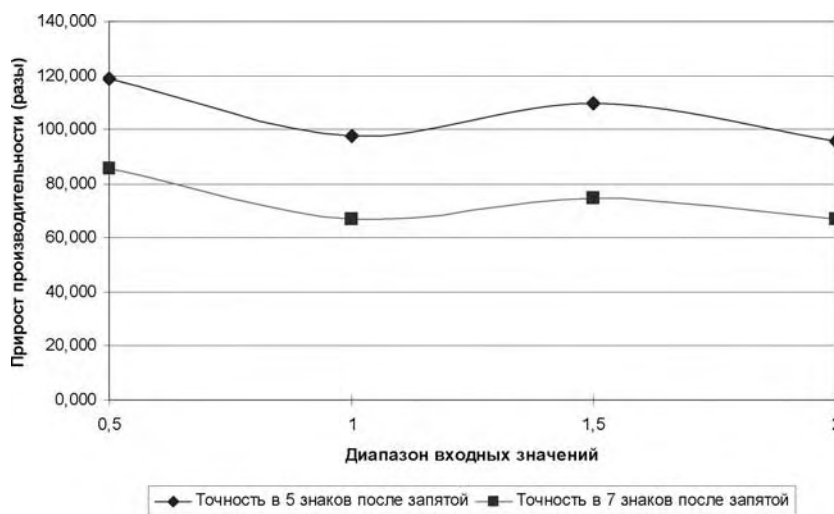


Рис. 7. Прирост производительности при использовании аппаратного модуля

ний пользователя. Пусть пользователем задан диапазон для входных значений от нуля до двух включительно, точность в пять знаков после запятой и выбран формат с фиксированной запятой. В качестве платформы была выбрана ПЛИС FPGA Xilinx Spartan3AN.

На основе приведенной экспертной информации можно предложить следующий вариант реализации модуля доступа (рис. 6).

Следует обратить внимание на использование механизмов утверждений [3, 4]. Они позволяют, с одной стороны, контролировать экспертные утверждения об использовании функции, с другой – вхождение полученных результатов в допустимый диапазон. Кроме того, показанный исходный код содержит дополнительную проверку, гарантирующую сравнение результатов работы сопроцессора и ожидаемых результатов с учетом заданной точности.

РЕЗУЛЬТАТЫ РЕАЛИЗАЦИИ ЭКСПОНЕНЦИАЛЬНОЙ ФУНКЦИИ

Даже в случае использования неоптимизированной реализации сопроцессора время работы варианта из стандартной библиотеки превышает время работы аппаратного модуля в 90 раз в лучшем случае (рис. 7), что доказывает эффективность предложенного подхода.

В качестве примера, который иллюстрирует важность качественной обработки экспертных данных, на диаграмме (рис. 8) показан прирост для точности в семь знаков. Следует отметить, что прирост аппаратных затрат в этом случае составил порядка 25 % относительно модуля с точностью в пять знаков.

ЗАКЛЮЧЕНИЕ

В данной статье предложен метод создания библиотеки математических функций для систем на ПЛИС.

Данный метод позволяет выполнять качественную оптимизацию систем на ПЛИС, которая основывается на двух важных факторах – экспертных знаниях о предметной области приложения и информации об используемой платформе.

Применение предложенного метода позволяет повысить производительность, что доказывают полученные результаты экспериментов.

Разработка библиотеки, обеспечивающей поддержку перечисленных платформ, сейчас ведется на базе лаборатории «FPGA-технологий и диагностики КС» ДонНТУ.

СПИСОК ЛИТЕРАТУРЫ

1. MicroBlaze Processor Reference Guide [Электронный ресурс] / XILINX. – Электрон. дан. – Режим доступа: http://www.xilinx.com/support/documentation/sw_manuals/edk63i_mb_ref_guide.pdf, свободный. – Загл. с экрана. – Англ.
2. Nios II Processor Reference Handbook [Электронный ресурс] / Altera Corporation. – Электрон. дан. – Режим доступа: http://www.altera.com/literature/hb/nios2/n2srcu_nii5v1.pdf, свободный. – Загл. с экрана. – Англ.
3. Programming languages – C : ISO/IEC 9899:1999(E) – ISO/IEC 9899/1999(E). – [Чинний від 01.12.1999]. – ISO/IEC, 1999. – 538 р. – (Міжнародний стандарт).
4. Programming languages – C++ : ISO/IEC 14882:2003 – ISO/IEC 14882:2003. – [Чинний від 01.12.2003]. – ISO/IEC, 2003. – 650 р. – (Міжнародний стандарт).
5. Jerraya, A. A. Hardware/Software Interface Codesign for Embedded Systems / A. A. Jerraya, W. Wolf // Computer. – 2005. – № 38. – P. 63–69.
6. Объектно-ориентированный анализ и проектирование с примерами приложений / [Г. Буч, Р. Максимчук,

М. Энгл и др.]. – [3-е изд.]. – М. : Вильямс», 2008. – 720 с. : ил.

7. Cyclone II Device Handbook Volume 1 [Электронный ресурс] / Altera Corporation. – Электрон. дан. – Режим доступа: http://www.altera.com/literature/hb/cyc2/cyc2_cii5v1.pdf, свободный. – Загл. с экрана. – Англ.
8. Avalon Interface Specifications [Электронный ресурс] / Altera Corporation. – Электрон. дан. – Режим доступа: http://www.altera.com/literature/manual/mnl_avalon_spec.pdf, свободный. – Загл. с экрана. – Англ.
9. Ильин, В. А. Математический анализ. Начальный курс / Ильин В. А., Садовничий В. А., Сендович Б. Ч. – [2-е изд., перераб.]. – М. : Изд-ва МГУ, 1985. – 662 с.

Надійшла 10.11.2010

Зінченко Ю. Є., Гриценко А. О., Зеленьова І. Я., Войтов Г. В.

ПРОГРАМНО-АППАРАТНА БІБЛІОТЕКА МАТЕМАТИЧНИХ ФУНКЦІЙ ДЛЯ СИСТЕМ НА ПЛІС

У статті пропонується метод створення бібліотеки математичних функцій для систем на ПЛІС (SoPC) і набору необхідних для роботи з нею апаратно-програмних засобів. Запропонована бібліотека надає можливість варіювати структурою системи на ПЛІС відповідно до вимог користувача та доступних апаратних ресурсів.

Ключові слова: ПЛІС, системи на ПЛІС, процесор, периферійний модуль, бібліотека.

Zinchenko Y. E., Grytsenko A. A., Zelenyova I. J., Voytov G. V.

HARDWARE-SOFTWARE MATH LIBRARY FOR SOPC

The authors propose a method of creation of a math library for SoPC and a set of the required tools. Using this library, it is possible to vary the SoPC structure according to the user's requirements and available hardware resources.

Key words: FPGA, SOPC, CPU, peripheral module, library.

УДК 004.416.03

Поздняков А. А.¹, Пархоменко А. В.², Тамрас Н. И.³, Чижик Е. В.⁴

¹Директор по консалтингу ООО «Бриг-Ритейл»

²Канд. техн. наук, доцент Запорожского национального технического университета

³Инженер-технолог ОАО «Мотор Сич»

⁴Инженер-программист ОАО «Мотор Сич»

АВТОМАТИЗИРОВАННОЕ УПРАВЛЕНИЕ ПРОЕКТОМ ВНЕДРЕНИЯ ПРОГРАММНОЙ СИСТЕМЫ

В статье приведены результаты интеграции современных программных продуктов с целью эффективного управления проектом внедрения системы SAP ERP.

Ключевые слова: методология внедрения ERP-системы, система управления проектом, интеллект-карта, иерархическая структура работ, интеграция программного обеспечения.

ВВЕДЕНИЕ

Одной из известных концепций построения информационной системы предприятия на современном этапе являются ERP-системы. Enterprise Resour-

ce Planning – это методология эффективного планирования и управления всеми ресурсами предприятия, которые необходимы для осуществления продаж, производства, закупок и учета при исполнении заказов

клиентов в сферах производства, дистрибуции и оказания услуг. Мировой опыт свидетельствует, что умело выбранная и внедренная ERP-система существенно улучшает управляемость предприятием и повышает эффективность его работы [1].

1. АКТУАЛЬНОСТЬ И ПРОБЛЕМЫ ВНЕДРЕНИЯ ERP-СИСТЕМ

В декабре 2009 года руководством ОАО «Мотор Сич» было принято решение автоматизировать деятельность предприятия по процессам закупок и хранения материалов, продаж готовой продукции, ведения бухгалтерского, налогового учета путем внедрения функциональных модулей системы SAP ERP (R/3 4.0b): «Финансы» (Finance, FI), «Управление материальными потоками» (Materials Management, MM), «Продажи и дистрибуция» (Sales and Distribution, SD) консалтинговой компанией BrigRetail.

Выбор предприятием данной системы обусловлен тем, что R/3 имеет мощный функционал и наибольшее количество внедрений среди ERP-систем на территории стран СНГ и в Украине. Наличие широкого спектра модульных приложений, позволяющих использовать их совместно, а также в комбинации с внешними элементами, открывает большие возможности по реализации операций в области финансов, логистики и управления персоналом [2].

Внедрение на предприятии такой системы, как R/3, дает возможность объединить производство, сбыт, бухгалтерский учет и учет затрат в единое целое, способствует рационализации производства, предоставляет возможность оперативного получения производственно-экономических данных и позволяет точно планировать и управлять производственными процессами.

Как известно, внедрение ERP-системы требует тщательного планирования в рамках длительного (от трех месяцев до нескольких лет) и дорогостоящего

проекта. Основные сложности внедрения возникают по следующим причинам: слабая поддержка проектов со стороны владельцев предприятий; нежелание предоставлять конфиденциальную информацию; недостаточное инвестирование в обучение персонала; недоработанность политики занесения и поддержки актуальности данных в ERP.

Тем не менее, проблемы, возникающие при внедрении подобных систем, уже достаточно хорошо изучены, и в настоящее время созданы эффективные методики их решения, объединенные в соответствующих стандартах (методологиях). Методологии внедрения обычно разрабатываются ведущими производителями информационных систем с учетом особенностей их программных продуктов, а также сферы внедрения.

Методология ASAP – это структурный подход, рекомендуемый компанией SAP AG при внедрении своей системы SAP ERP. На всех этапах методологии ASAP (подготовка проекта, концептуальное проектирование, реализация, окончательная подготовка, запуск и поддержка системы) активно решаются задачи, связанные с планированием и управлением проектом [3].

2. СИСТЕМЫ УПРАВЛЕНИЯ ПРОЕКТАМИ (СУП)

2.1. Анализ функциональных возможностей современных СУП

Для автоматизированного управления проектами сегодня широко применяются системы управления проектами. СУП позволяют обеспечить информационную поддержку жизненного цикла проектов, эффективное планирование и управление ходом работ, соответствие заранее определенным стандартам и требованиям [4].

Все системы управления проектами делятся на два класса – старший и средний (рис. 1). Принадлеж-

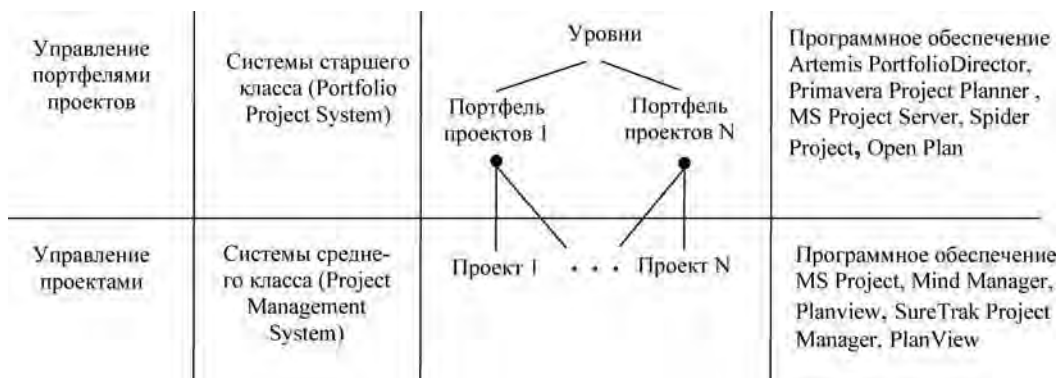


Рис. 1. Уровни управления проектами и используемое программное обеспечение

ность программной системы к тому или иному уровню определяется возможностью группировать проекты в портфели [5].

Как показал обзор [6, 7], практически все популярные инструменты «доросли» до того уровня, когда их возможности почти не отличаются. Поэтому при выборе СУП аналитики советуют принимать во внимание три вещи: зрелость существующих в компании практик управления проектами; типы инвестиций, проектов и работ, которые будут выполняться; специфические отраслевые требования и необходимость в дополнительной функциональности.

2.2. Выбор системы управления проектом

Для того, чтобы начать работу над проектом внедрения программной системы, необходимо сформулировать цели внедрения, подготовить стратегию внедрения, масштабировать и оценить предстоящий проект. Авторы предлагают руководителю проекта вначале сгенерировать свои идеи, структурировать и визуально их представить при помощи интеллект-карты (карты ума, ментальной карты). Интеллект-карта – это эффективный инструмент структурирования и анализа информации, позволяющий ускорить разработку бизнес-проектов [8].

Из существующих систем для визуального представления мыслей человека в виде интеллект-карты была выбрана программа Mindjet MindManager, которая представляет собой удобный инструмент, облегчающий планирование и контроль проектов [9]. Диаграмма идеи, которая создается с помощью MindMan-

ager, имитирует ход мысли автора, а также предлагает возможности интерактивной работы, в том числе встраивание в диаграмму разного рода данных, гиперссылок, инструментов коллективной работы с информацией и др.

Для последующей более детальной работы над проектом была выбрана система Microsoft Project, которая предлагает качественную реализацию стандартных функций управления проектом за умеренную цену [10]. Достоинствами этой системы являются простота эксплуатации, интуитивность управления и настройки. Между выбранными системами легко осуществляется передача данных, причем в обоих направлениях.

3. МЕТОДИКА УПРАВЛЕНИЯ ПРОЕКТОМ

3.1. Совместное применение MindManager и MS Project для планирования и управления проектом

Для эффективного планирования проекта внедрения системы SAP ERP (R/3) на ОАО «Мотор Сич» авторами предложена следующая методика.

1-й этап. Выполнить создание ментальной карты, которая отображает укрупненный план внедрения системы. На ней создаются: Центральная тема (Внедрение системы R/3) – соответствует понятию Проект в Microsoft Project, подтемы 1-го уровня (Подготовка проекта, Концептуальное проектирование, Реализация) – соответствуют Иерархической Структуре Работ (ИСР), подтемы 2-го уровня – соответствуют Итоговым задачам (рис. 2).

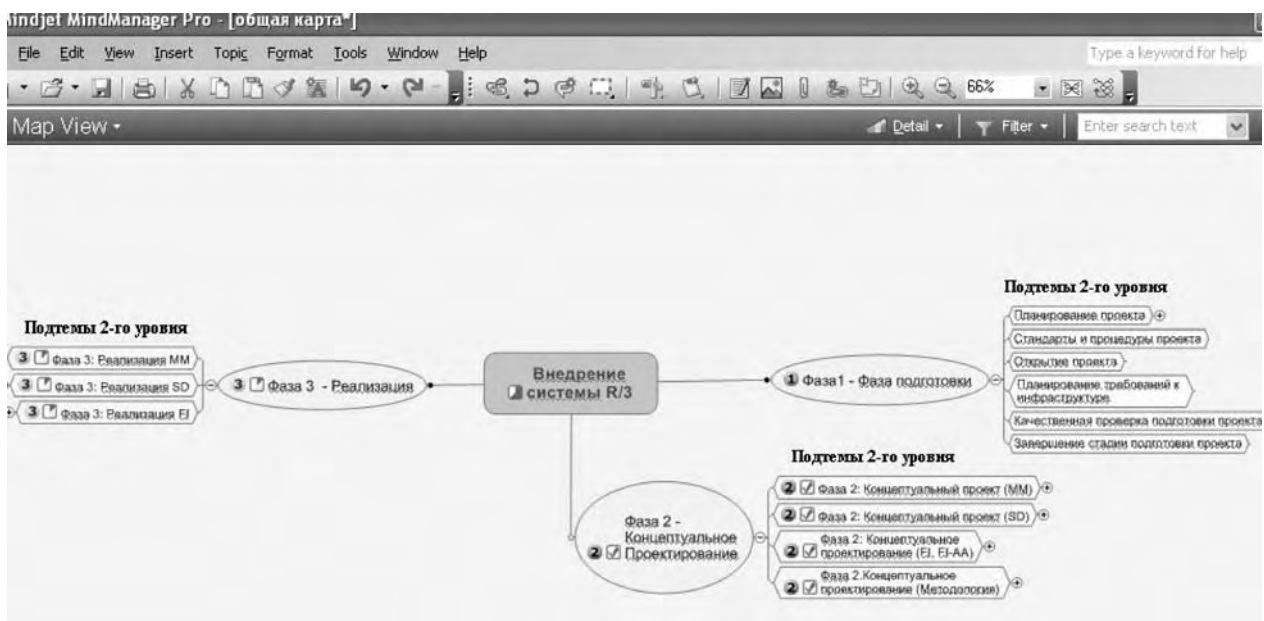


Рис. 2. Разработка ментальной карты в MindManager

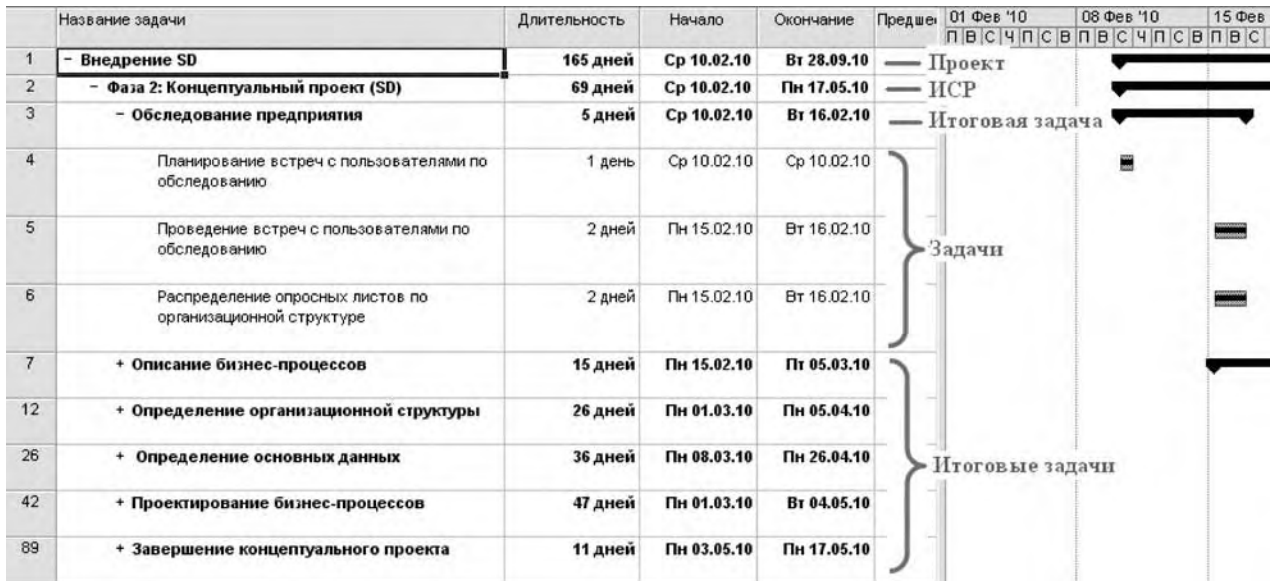


Рис. 3. Результат передачи данных из MindManager в MS Project

2-й этап. Экспорт данных из MindManager в MS Project. Все, что присутствует на ментальной карте, передается в MS Project в виде: Проект, ИСР, Итоговые задачи (рис. 3).

3-й этап. Произвести в MS Project детализацию укрупненного плана, т. е. разбить Итоговые задачи на Задачи. Перечислить все ресурсы, которые будут задействованы в проекте, определить их тип (трудовой, материальный), сгруппировать их. Для задач наиболее низкого уровня определить длительность, назначить для них ресурсы. Установить связи между задачами и ограничения на задачи. Провести анализ и оптимизацию плана. Выполнить подготовку отчетов.

4-й этап. Если есть необходимость наглядно представить данные, например, запаздывающие задачи, то создать соответствующую выборку таких задач и передать их обратно в MindManager, где в свою очередь будет создана наглядная ментальная карта. При необходимости сохранить график ресурсов в файл GIF и прикрепить его к ментальной карте.

Предложенная методика базируется на интеграции выбранных программных продуктов, что обеспечивает необходимый функционал и представляет хорошую альтернативу более дорогостоящим и сложным в использовании системам управления проектами.

3.2. Разработка программного обеспечения для расширения функциональных возможностей MS Project

Как известно, основная задача руководителя проекта – обеспечить выполнение работ в установлен-

ные сроки, в рамках выделенных средств и с соответствующим качеством [11].

Для успешной реализации проекта необходимо не только составить подробный план, учитывающий требования по использованию ресурсов, но и оперативно контролировать исполнение работ, отслеживать сроки и соблюдать ограничения по объему финансирования, своевременно внося необходимые коррективы. Это и является сутью процесса управления.

Для упрощения этого процесса в среде MS Project были разработаны две формы: «Работа с задачами» и «Работа с ресурсами». Необходимость создания этих форм обуславливается тем, что руководителю проекта ежедневно приходится осуществлять анализ следующих данных: какие задачи выполнены, планируются, запаздывают, при этом всегда под контролем должны быть критические задачи. Также необходимо контролировать выполнение задач группами ресурсов или отдельными ресурсами и т. д. Поэтому, для экономии времени руководителя проекта и избавления его от рутинной работы (всевозможного применения фильтров, сортировок, группировок), авторами были разработаны формы, на которых отображены наиболее часто используемые команды. Разработанные формы обладают интуитивно понятным интерфейсом и используются для получения объективной оценки хода выполнения проекта.

Для разработки форм в MS Project использовался VBA (Visual Basic for Applications), являющийся инструментальным средством разработки приложений в среде основных компонентов Office.



Рис. 4. Форма «Работа с задачами»

Разработанная форма «Работа с задачами» (рис. 4) позволяет пользователю:

- в блоке «Задачи» получить сведения о ходе выполнения всех задач проекта;
- в блоке «Критические задачи» получить сведения о ходе выполнения критических задач;
- в блоке «Группы ресурсов» получить сведения о задачах, выполняемых определенной группой ресурсов;
- в блоке «Отчеты» просмотреть и распечатать сведения о задачах.

Разработанная форма «Работа с ресурсами» позволяет пользователю:

- в блоке «Ресурсы» получить сведения о ресурсах, задействованных в выполнении проекта;
- в блоке «Группа ресурсов» получить сведения о выполнении задач конкретной группы ресурсов;
- в блоке «Отчеты» автоматизировать процесс формирования отчетов.

3.3. Возможности интеграции СУП с системой SAP ERP (R/3)

Необходимо отметить, что в составе системы R/3 имеется прикладной модуль «Управление проектами» (Project System, PS), который поддерживает планирование, управление и мониторинг долгосрочных проектов с высоким уровнем сложности.

В случае дальнейшего внедрения на предприятии данного модуля авторами предлагается совместное использование программ MindManager, MS Project для расширения функциональности модуля PS. Это связано с тем, что модуль PS не поддерживает всех функций календарного планирования, однако позволяет вводить уже разработанный план проекта или загружать, например, из MS Project (рис. 5).

После получения при помощи модуля PS фактических затрат по проекту осуществляется их обратная передача в MS Project для получения отклонения между плановыми и фактическими показателями. На основе этих сведений выполняется анализ возможного влияния отклонений в выполненных объемах работ на ход реализации проекта в целом и выработка соответствующих управленческих решений.

ВЫВОДЫ

В работе предложена методика эффективного управления проектом внедрения системы SAP ERP (R/3), которая базируется на методологии ASAP, а также принципе открытости программных систем. Интеграция выбранных программных продуктов обеспечивает необходимый функционал и представляет хорошую альтернативу более дорогостоящим и сложным в использовании системам управления проектами. Разработанные в среде MS Project формы «Работа с задачами» и «Работа с ресурсами» могут использоваться для получения объективной оценки хода выполнения проекта. Практическая ценность разработки заключается в экономии времени руководителя проекта и избавлении его от рутинной работы. Перспективным является дальнейшее совместное применение модуля PS системы SAP ERP совместно с программами MindManager и MS Project.

Автоматизированное управление проектом внедрения системы R/3 на предприятии «Мотор Сич» позволило контролировать выполнение проекта в установленные сроки, с учетом утвержденного бюджета и с соответствующим качеством.

ПЕРЕЧЕНЬ ССЫЛОК

1. Поздняков, О. А. Особенности создания системы управления предприятием на основе MYSAP ERP / О. А. Поздняков и др. // *Материалы Международной НТК «Компьютерное моделирование и интеллектуальные системы».* – Запорожье : ЗНТУ, 2007. – С. 240–247.
2. Кобыляцкий, А. Г. Адаптация системы R/3 для решения задач пользователя / А. Г. Кобыляцкий и др. // *Радиоэлектроника, информатика, управление.* – № 1, 2009. – С. 124–128.
3. Кале, В. Внедрение SAP R/3. Руководство для менеджеров и инженеров : пер. с англ. П. А. Панов / В. Кале. – М. : Компания АйТи, 2006. – 511 с.

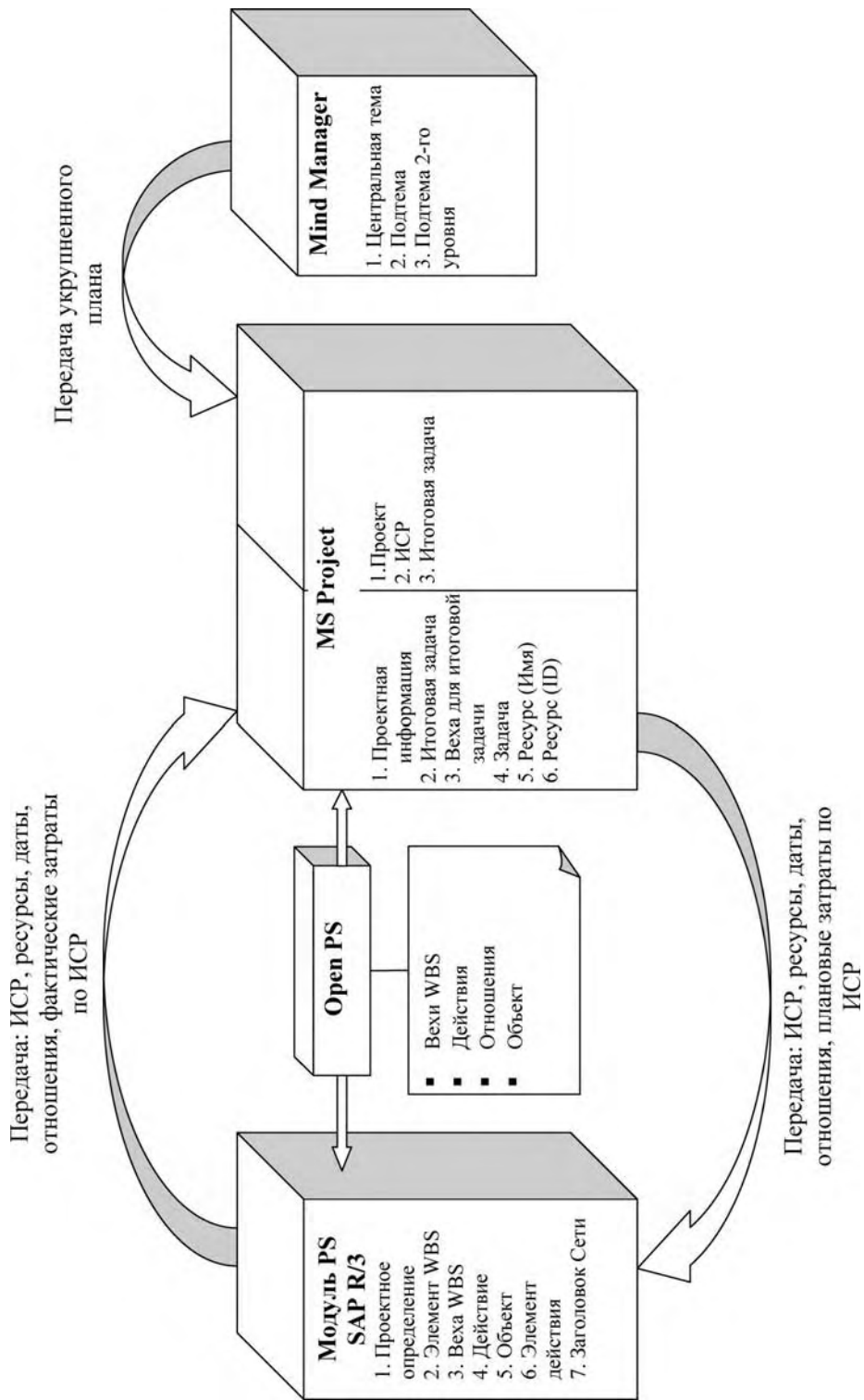


Рис. 5. Взаимодействие программных продуктов

4. *Вязовой, В.* Системы управления проектами [Электронный ресурс] / Валерий Вязовой. – Электрон. дані. – Режим доступа: <http://www.cfin.ru/software/project/pms-review.shtml>, вільний. – Назва з екрана.
5. *Матвеев, А. А.* Модели и методы управления портфелями проектов / А. А. Матвеев, Д. А. Новиков, А. В. Цветков. – М. : ПМСОФТ, 2005. – 206 с.
6. Обзор современного программного обеспечения управления проектами [Электронный ресурс]. – Режим доступа: <http://www.bibliofond.ru/view.aspx?id=66417>, свободный. – Назв. с экрана.
7. *Кухар, А.* Инструментальная поддержка PPM / А. Кухар // Компьютерное обозрение. – 2009. – № 36. – С. 26–29.
8. *Мюллер, Х.* Составление ментальных карт. Метод генерации и структурирования идей / Хорст Мюллер. – М. : Омега-Л, 2007. – 356 с.
9. *Копыл, В. И.* Карты ума. MindManager / В. И. Копыл – Минск : Харвест, 2007. – 64 с.
10. *Кудрявцев, Е. М.* Microsoft Project. Методы сетевого планирования и управления проектом / Е. М. Кудрявцев. – М. : ДМК Пресс, 2005. – 240 с.
11. *Арчибальд, Р. Д.* Как улучшить управление проектами в организации / Рассел Д. Арчибальд // Методы менеджмента качества. – № 8, 2004. – С. 17–23.

Надійшла 14.10.2010

Поздняков А. А., Пархоменко А. В., Тамрас Н. И., Чижик О. В.

АВТОМАТИЗОВАНЕ УПРАВЛІННЯ ПРОЕКТОМ ВПРОВАДЖЕННЯ ПРОГРАМНОЇ СИСТЕМИ

У статті наведено результати інтеграції сучасних програмних продуктів з метою ефективного управління проектом впровадження системи SAP ERP.

Ключові слова: методологія впровадження ERP-системи, система управління проектом, інтелект-карта, ієрархічна структура робіт, інтеграція програмного забезпечення.

Pozdnyakov A. A., Parkhomenko A. V., Tamras N. I., Chizhik O. V.

AUTOMATIZED MANAGEMENT OF PROGRAM SYSTEM INTRODUCTION PROJECT

The results of modern software integration for the purpose of SAP ERP-system introduction project effective management are presented in this paper.

Key words: ERP-systems introduction methodology, project management system, mind map, Work Breakdown Structure, software integration.

УДК 658.512.011:681.326:519.713

Хаханов В. И.¹, Чумаченко С. В.¹, Литвинова Е. И.², Гузь О. А.³

¹Д-р техн. наук, профессор Харьковского национального университета радиозлектроники

²Канд. техн. наук, доцент Харьковского национального университета радиозлектроники

³Канд. техн. наук, доцент Донецкой академии автомобильного транспорта

МУЛЬТИПРОЦЕССОР ДЛЯ АНАЛИЗА ИНФОРМАЦИОННОГО ПРОСТРАНСТВА

Предлагается инфраструктура быстродействующего мультипроцессора параллельного анализа информации, представленной в виде аналитических, графовых и табличных форм ассоциативных отношений, для поиска, распознавания и принятия решений в n -мерном векторном дискретном пространстве. Рассматриваются векторно-логические процесс-модели актуальных прикладных задач, качество решения которых оценивается введенной интегральной неарифметической метрикой взаимодействия булевых векторов.

Ключевые слова: мультипроцессор, анализ информации, логическое ассоциативное отношение, процесс-модель.

ВВЕДЕНИЕ

Идея исследования заключается в том, чтобы убрать из компьютера «тяжеловесную» арифметику и трансформировать освободившиеся ресурсы в (мозгоподобную) инфраструктуру ассоциативной логики, моделирующей функциональность мозга, которая позволяет опытному человеку ежеминутно принимать правильные решения. Мозг и компьютер имеют единую технологическую основу в виде примитивных логических операций: and, or, not, xor. По мере приобретения опыта мозг и компьютер создают более сложные функциональные пространствен-

но-временные логические преобразователи, использующие упомянутые выше примитивные операции. Специализация компьютерного изделия, ориентированная на использование только логических операций, дает возможность приблизиться к ассоциативно-логическому мышлению человека и тем самым существенно ($\times 100$) повысить быстродействие решения неарифметических задач. Исключение арифметических операций, использование параллелизма алгебры векторной логики, мультипроцессорность архитектуры создают эффективную инфраструктуру, которая объединяет математическую и технологическую

культуру для решения прикладных задач. Мозгоподобность мультипроцессорной цифровой системы на кристалле есть концепция создания архитектуры и моделей вычислительных процессов для реализации свойственных мозгу неарифметических ассоциативно-логических функциональностей на современной цифровой платформе путем использования векторных логических операций и критериев в задачах поиска, распознавания и принятия решений. Рыночная привлекательность логического ассоциативного мультипроцессора (Logical Associative MultiProcessor – LAMP) определяется тысячами старых и новых логических по своей природе задач, которые в настоящее время решаются не эффективно на избыточных универсальных компьютерах с мощным арифметическим процессором. Вот некоторые актуальные для IT-рынка проблемы:

1. Анализ, синтез и коррекция синтаксических и семантических языковых конструкций (реферирование, исправление ошибок, оценивание качества текстов).

2. Распознавание видео- и аудио-образов путем их представления векторными моделями существенных параметров в дискретном пространстве.

3. Сервисное обслуживание сложных технических изделий и восстановление работоспособности в процессе их функционирования.

4. Тестирование знаний и экспертное обслуживание объектов или субъектов для определения их валидности.

5. Идентификация объекта или процесса для принятия решения в условиях неопределенности.

6. Точный поиск заданной вектором параметров информации в Internet.

7. Целеуказание в истребителе или в автоматической системе посадки лайнера, работающей в реальном микросекундном диапазоне времени.

8. Разведение объектов во времени и в пространстве для диспетчерской службы аэропорта или оптимизация инфраструктуры городского транспорта в целях исключения коллизий.

Практически все упомянутые задачи решаются в реальном масштабе времени, являются изоморфными по логической структуре процесс-моделей, использующих совокупность взаимосвязанных ассоциативных таблиц. Для их решения необходима быстродействующая и специализированная аппаратная платформа (LAMP), ориентированная на параллельное выполнение процедур поиска, распознавания и принятия решений, оцениваемых путем использования интегрального неарифметического критерия качества.

Целью работы является существенное ($\times 100$) повышение быстродействия процедур поиска, распознавания и принятия решений путем мультипроцессорной и параллельной реализации ассоциативно-логических векторных операций для анализа графовых и табличных структур данных в дискретном булевом пространстве без использования арифметических операций.

Для достижения цели необходимо решить следующие задачи:

1) разработать неарифметическую метрику оценивания ассоциативно-логических решений;

2) сформировать структуры данных и процесс-модели решения актуальных задач;

3) создать архитектуру логического ассоциативного мультипроцессора;

4) определить практическое использование LAMP.

Сущность исследования заключается в разработке инфраструктуры экспертного обслуживания запросов в реальном масштабе времени, интегрирующей мультипроцессорную систему на кристалле с аппаратом ассоциативно-логических структур данных для получения детерминированного решения, валидность (состоятельность) которого оценивается неарифметическим интегральным критерием качества взаимодействия запроса с заданным дискретным пространством.

Объект исследования – инфраструктура поиска, распознавания и принятия решений в дискретном булевом пространстве на основе использования алгебры векторной логики, мультипроцессорной платформы анализа ассоциативно-логических структур данных и неарифметического интегрального критерия качества.

Предмет исследования – ассоциативно-логические структуры данных и процесс-модели поиска, распознавания и выбора решения на основе неарифметического интегрального критерия качества путем использования мультипроцессорной системы на кристалле, оперирующей векторными логическими операциями.

Источники: аппаратная платформа ассоциативно-логического анализа информации [1–4]; ассоциативно-логические структуры данных для решения информационных задач [5–8]; модели и методы дискретного анализа и синтеза [9–12]; мультипроцессорные средства решения информационно-логических задач [13–19]; мозгоподобные и интеллектуальные логические вычисления [20–25].

1. ИНТЕГРАЛЬНАЯ МЕТРИКА ОЦЕНИВАНИЯ РЕШЕНИЯ

Инфраструктура мозгоподобного мультипроцессора включает модели, методы и ассоциативно-логичес-

кие структуры данных, ориентированные на аппаратную поддержку процессов поиска, распознавания и принятия решений [22–24] на основе векторных неарифметических операций. Оценка решения задачи определяется векторно-логическим критерием качества взаимодействия запроса (вектора m) с системой ассоциативных векторов (ассоциаторов), в результате которого сгенерируется конструктивный ответ в виде одного или нескольких ассоциаторов, а также пока еще численной характеристики степени принадлежности (функции качества) входного вектора m к найденному решению: $\mu(m \in A)$. Входной вектор $m = (m_1, m_2, \dots, m_i, \dots, m_q)$, $m_i \in \{0, 1, x\}$ и матрица A_i ассоциаторов $A_{ij}, (A_{ij} \in A_i \in A) = \{0, 1, x\}$ имеют одинаковую размерность, равную q . Далее степень принадлежности m -вектора к вектору A будет обозначаться как $\mu(m \in A)$.

Существует 5 типов теоретико-множественного (логического) Δ -взаимодействия двух векторов $m \cap A$, определенных на рис. 1. Они формируют все примитивные варианты реакции обобщенной ПРП-системы (Поиска, Распознавания и Принятия решения) на входной вектор-запрос. В технологической отрасли знаний – технической диагностике (Design & Test) – указанная последовательность действий изоморфна маршруту: поиск дефектов, их распознавание, принятие решения на восстановление работоспособности. Все три стадии технологического маршрута нуждаются в метрике оценивания решений для выбора оптимального варианта.

Определение. Интегральная теоретико-множественная метрика для оценивания качества запроса есть функция взаимодействия многозначных векторов $m \cap A$, которая определяется средней суммой трех нормированных параметров: кодового расстояния $d(m, A)$, функции принадлежности $\mu(m \in A)$ и функции принадлежности $\mu(A \in m)$:

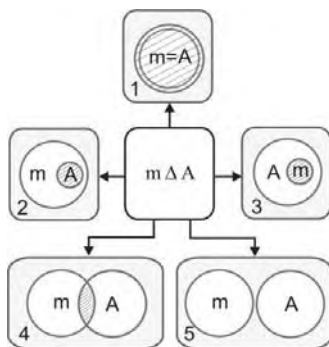


Рис. 1. Результаты пересечения двух векторов

$$Q = \frac{1}{3} [d(m, A) + \mu(m \in A) + \mu(A \in m)], d(m, A) = \frac{1}{n} \left[n - \text{card} \left(m_i \bigcap_{i=1}^n A_i = \emptyset \right) \right];$$

$$\mu(m \in A) = 2^{\text{card}(m \cap A) - \text{card}(A)} \leftarrow \text{card}(m \cap A) = \text{card} \left(m_i \bigcap_{i=1}^n A_i = x \right) \& \text{card}(A) = \text{card} \left(\bigcup_{i=1}^n A_i = x \right);$$

$$\mu(A \in m) = 2^{\text{card}(m \cap A) - \text{card}(m)} \leftarrow \text{card}(m \cap A) = \text{card} \left(m_i \bigcap_{i=1}^n A_i = x \right) \& \text{card}(m) = \text{card} \left(\bigcup_{i=1}^n m_i = x \right). \quad (1)$$

Пояснения. Нормирование параметров позволяет оценить уровень взаимодействия векторов в интервале $[0, 1]$. Если зафиксировано предельное максимальное значение каждого параметра, равное 1, то векторы равны между собой. Минимальная оценка, $Q = 0$, фиксируется в случае полного несовпадения векторов по всем n координатам. Если мощность пересечения $m \cap A = m$ равна половине пространства вектора A , то функции принадлежности и качества соответственно равны:

$$\mu(m \in A) = \frac{1}{2}; \mu(A \in m) = 1; d(m, A) = 1;$$

$$Q(m, A) = \frac{5}{2 \times 3} = \frac{5}{6}.$$

Аналогичное значение будет иметь параметр Q , если мощность пересечения $m \cap A = A$ равна половине пространства вектора m . Если мощность пересечения $\text{card}(m \cap A)$ равна половине мощностей пространств векторов A и m , то функции принадлежности имеют следующие значения:

$$\mu(m \in A) = \frac{1}{2}; \mu(A \in m) = \frac{1}{2};$$

$$d(m, A) = 1;$$

$$Q(m, A) = \frac{4}{2 \times 3} = \frac{4}{6} = \frac{2}{3}.$$

Следует заметить, если пересечение двух векторов равно пустому множеству, то степень двойки от символа «пусто» равна нулю: $2^{\text{card}(m \cap A) = \emptyset} = 2^\emptyset = 0$. Это действительно означает, что количество общих точек при пересечении двух пространств равно нулю.

Цель введения векторно-логического критерия качества решения заключается в существенном повышении быстродействия при подсчете качества Q взаимодействия компонентов m и A при анализе ассоциативных структур данных путем использования только векторных логических операций. Арифметический критерий (1) без усреднения функций при-

надлежности и кодового расстояния можно трансформировать к виду

$$Q = d[m, A_{i(j)}] + \mu[m \in A_{i(j)}] + \mu[A_{i(j)} \in m],$$

$$d(m, A_{i(j)}) = \text{card}[m \oplus_{i(j)=1}^{n(m)} A_{i(j)} = 1];$$

$$\mu(m \in A_{i(j)}) = \text{card}[A_{i(j)} = 1] - \text{card}[m \wedge_{i(j)=1}^{n(m)} A_{i(j)} = 1];$$

$$\mu(A_{i(j)} \in m) = \text{card}[m = 1] - \text{card}[m \wedge_{i(j)=1}^{n(m)} A_{i(j)} = 1]. \quad (2)$$

Первый компонент, составляющий критерий, формирует степень несовпадения n -мерных векторов – кодовое расстояние, путем выполнения операции хог, второй и третий определяют степень непринадлежности результата конъюнкции к числу единиц каждого из двух взаимодействующих векторов. Понятия принадлежности и непринадлежности являются взаимодополняющими, но в данном случае технологичнее вычислять непринадлежность. Таким образом, идеальный критерий качества равен нулю, когда два вектора равны между собой. Оценка качества взаимодействия двух двоичных векторов убывает по мере роста критерия от 0 к 1. Чтобы окончательно уйти от арифметических операций при подсчете уже век-

торного критерия качества, необходимо выражения (2) преобразовать к виду

$$Q = d(m, A) \vee \mu(m \in A) \vee \mu(A \in m),$$

$$d(m, A) = m \oplus A;$$

$$\mu(m \in A) = A \in \overline{m \wedge A};$$

$$\mu(A \in m) = m \in \overline{m \wedge A}. \quad (3)$$

Здесь критерии представлены уже не числами, а векторами, которые оценивают взаимодействие компонентов m, A . При этом увеличение числа нулей в трех векторах качества повышает критерий, а наличие единиц индицирует ухудшение качества взаимодействия. Для сравнения оценок необходимо определять мощность единиц в каждом векторе без выполнения операций суммирования. Это можно сделать с помощью регистра [4] (рис. 2), который позволяет за один такт выполнить сдвиг влево и уплотнить все единичные координаты n -разрядного двоичного вектора.

После процедуры сжатия номер правого единичного бита уплотненной серии единиц формирует индекс качества взаимодействия векторов. Для двоичных наборов $m = (110011001100)$, $A = (000011110101)$ определение качества их взаимодействия по формулам (3) представлено в следующем виде (нулевые координаты отмечены точками):

m	1 1 . . 1 1 . . 1 1 . .
A 1 1 1 1 . 1 . 1
$m \wedge A$ 1 1 . . . 1 . .
$m \wedge A$	1 1 1 1 . . 1 1 1 . 1 1
$d(m, A) = m \oplus A$	1 1 1 1 1 . . 1
$\mu(A \in m) = m \wedge \overline{m \wedge A}$	1 1 1 . . .
$\mu(m \in A) = A \wedge \overline{m \wedge A}$ 1 1 . . . 1
$Q = d(m, A) \vee \mu(m \in A) \vee \mu(A \in m)$	1 1 1 1 1 . . 1
$Q(m, A) = (6/12)$	1 1 1 1 1 1

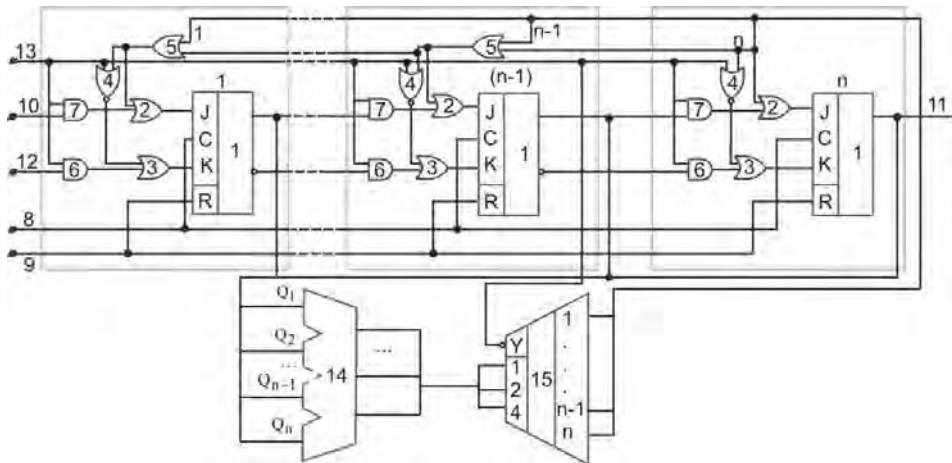


Рис. 2. Регистр сдвига и уплотнения единиц

Здесь сформирована не только оценка взаимодействия векторов, равная $Q(m, A) = (6/12)$, но, что самое главное, единичные координаты строки $Q = d(m, A) \vee \mu(m \in A) \vee \mu(A \in m)$ идентифицируют все существенные переменные, по которым имеется некачественное взаимодействие векторов. Для сравнения двух решений, полученных в результате логического анализа, используются сжатые векторы качества Q , над которыми выполняется процедура, включающая следующие векторные операции:

$$Q(m, A) = \begin{cases} Q_1(m, A) \leftarrow \text{or}[Q_1(m, A) \wedge Q_2(m, A) \oplus Q_1(m, A)] = 0; \\ Q_2(m, A) \leftarrow \text{or}[Q_1(m, A) \wedge Q_2(m, A) \oplus Q_1(m, A)] = 1. \end{cases} \quad (4)$$

Вектор-бит or -оператор девекторизации формирует двоичное битовое решение на основе применения логической операции or к n разрядам вектора существенных переменных критерия качества. Схематическое решение процедуры выбора

$$Q = \begin{cases} Q_1 \leftarrow Y = 0 \\ Q_2 \leftarrow Y = 1 \end{cases}$$

и аналитическая процесс-модель имеют три операции, которые представлены на рис. 3.

Для двоичных векторов, представляющих собой критерии качества, ниже выполнена процедура выбора лучшего их них на основании выражения, представленного в (4):

$Q_1(m, A) = (6,12)$	1 1 1 1 1 1
$Q_2(m, A) = (8,12)$	1 1 1 1 1 1 1
$Q_1(m, A) \wedge Q_2(m, A)$	1 1 1 1 1 1
$Q_1(m, A) \oplus Q_1(m, A) \wedge Q_2(m, A)$
$Q(m, A) = Q_1(m, A)$	1 1 1 1 1 1

Векторные логические критерии качества взаимодействия ассоциативных наборов позволяют получать оценку поиска, распознавания и принятия решения с высоким быстродействием логических параллельных операций, что особенно существенно для критических систем реального времени.

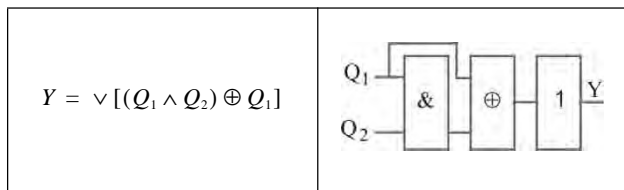


Рис. 3. Процесс-модель выбора решения

2. ПРОЦЕСС-МОДЕЛЬ ПОИСКА, РАСПОЗНАВАНИЯ И ПРИНЯТИЯ РЕШЕНИЯ

Метрика качества, представленная в (3), дает возможность оценивать близость пространственных объектов друг к другу, а также взаимодействие векторных пространств. Представленная ниже процесс-модель $P(m, A)$ сопровождается интегральным критерием качества, который оценивает не только кодовое расстояние по Хэммингу, но и эффективность взаимодействия объектов. Аналитическая запись обобщенной процесс-модели для выбора лучшего взаимодействия входного запроса m с системой логических ассоциативных отношений представлена в следующем виде:

$$P(m, A) = \min_{i=1}^n Q_i(m \Delta A_i) = \vee [(Q_i \wedge_{j=1, n}^{j \neq i} Q_j) \oplus Q_i] = 0;$$

$$Q(m, A) = (Q_1, Q_2, \dots, Q_i, \dots, Q_n);$$

$$A = (A_1, A_2, \dots, A_i, \dots, A_n);$$

$$\Delta = \{\text{and, or, xor, not, slc, nop}\};$$

$$A_i = (A_{i1}, A_{i2}, \dots, A_{ij}, \dots, A_{is});$$

$$A_{ij} = (A_{ij1}, A_{ij2}, \dots, A_{ijr}, \dots, A_{msq});$$

$$m = (m_1, m_2, \dots, m_r, \dots, m_q);$$

$$Q_i = d(m, A_i) \vee \mu(m \in A_i) \vee \mu(A_i \in m),$$

$$d(m, A_i) = m \oplus A_i;$$

$$\mu(m \in A_i) = A_i \wedge \overline{m \wedge A_i};$$

$$\mu(A_i \in m) = m \wedge \overline{m \wedge A_i}. \quad (5)$$

Комментарии: 1) функциональность $P(m, A)$ задает аналитическую модель вычислительного процесса в виде высказывания, минимизирующего интегральный критерий качества; 2) структуры данных представлены в виде вершин-таблиц графа $A = (A_1, A_2, \dots, A_i, \dots, A_m)$, логически взаимодействующих между собой; 3) вершина графа задается упорядоченной совокупностью вектор-строк ассоциативной таблицы $A_i = (A_{i1}, A_{i2}, \dots, A_{ij}, \dots, A_{is})$ явных решений, где строка $A_{ij} = (A_{ij1}, A_{ij2}, \dots, A_{ijr}, \dots, A_{msq})$ представляет собой истинное высказывание. Поскольку функционал, представленный в виде таблицы, не имеет постоянных во времени входных и выходных переменных, то данная структура отличается от последовательной машины фон Неймана, задаваемой конечными автоматами Мили и Мура. Равнозначность всех переменных в векторе $A_{ij} = (A_{ij1}, A_{ij2}, \dots, A_{ijr}, \dots, A_{msq})$ создает одинаковые условия их существования, что означает инвариантность решения задач прямой и обратной импликации

в пространстве $A_i \in A$. Ассоциативный вектор A_{ij} определяет собой явное решение, где каждая переменная задается в конечном, многозначном и дискретном алфавите $A_{ijr} \in \{\alpha_1, \alpha_2, \dots, \alpha_i, \dots, \alpha_k\} = \beta$. Взаимодействие $P(m, A)$, входного вектора-запроса $m = (m_1, m_2, \dots, m_r, \dots, m_q)$ с графом $A = (A_1, A_2, \dots, A_i, \dots, A_m)$, формирует множество решений с выбором лучшего из них по минимальному критерию качества:

$$P(m, A) = \min Q_i [m \wedge (A_1 \vee A_2 \vee \dots \vee A_i \vee \dots \vee A_m)].$$

Конкретное взаимодействие вершин графа между собой создает функциональность $A = (A_1, A_2, \dots, A_i, \dots, A_m)$, которая может быть оформлена в следующие структуры:

1) единственная ассоциативная таблица, содержащая все решения логической задачи в явном виде. Преимущество – максимальное быстродействие параллельного ассоциативного поиска решения по таблице. Недостаток – максимально высокая аппаратурная сложность размещения таблицы большой размерности;

2) древовидная (графовая) структура бинарных отношений между функциональными примитивами, каждый из которых формирует таблицу истинности для незначительного количества переменных. Преимущество – максимально низкая аппаратурная сложность решения задачи. Недостаток – минимальное быстродействие последовательного ассоциативного поиска решения по дереву;

3) компромиссная графовая структура логически понятных для пользователя отношений между примитивами, каждый из которых формирует таблицу истинности для логически сильно взаимосвязанных переменных. Преимущество – высокое быстродействие параллельного ассоциативного поиска решений по минимальному числу таблиц, составляющих граф, а также сравнительно невысокая аппаратурная сложность решения задачи. Недостаток – снижение быстродействия из-за последовательной логической обработки графовой структуры явных решений, найденных в таблицах. Разбиение одной таблицы (ассоциативной памяти) на k частей приводит к уменьшению аппаратных затрат, выраженных в компонентах (логических таблицах) (LUT – Look Up Table) программируемой логической матрицы [15, 16]. Каждая ячейка памяти создается с помощью четырех лутов. Учитывая, что ассоциативную матрицу можно представить квадратом со стороной n , то суммарные аппаратные затраты $Z(n)$ памяти для хранения данных и время $T(n)$ анализа логического ассо-

циативного графа функционально зависят от числа n разбиений таблицы или числа вершин графа:

$$Z(n) = k \times \frac{1}{4} \times \left(\frac{n}{k}\right)^2 + h = \frac{n^2}{4 \times k} + h, \quad (h = \{n, \text{const}\});$$

$$T(n) = \frac{4 \times k}{t_{clk}} + \frac{4}{t_{clk}} = \frac{4}{t_{clk}}(k + 1), \quad (t_{clk} = \text{const}). \quad (6)$$

Здесь h – затраты на общую схему управления системой ассоциативных памяти. Платой за уменьшение аппаратуры является снижение быстродействия обработки структуры памяти или увеличение периода анализа компонентов системы. Период обработки одной ассоциативной памяти представлен циклом из четырех синхроимпульсов. Число разбиений k пропорционально увеличивает количество тактов в худшем варианте последовательного соединения памяти. Слагаемое $\frac{4}{t_{clk}}$ задает время, необходимое для подготовки данных на входе системы, а также для их декодирования на выходе вычислительной структуры. Функциональные зависимости аппаратных затрат и времени анализа графа ассоциативных памяти от числа вершин или разбиений представлены на рис. 4.

Обобщенная функция эффективности графовой структуры от числа вершин

$$f[Z(n), T(n)] = Z(n) + T(n) = \left(\frac{n^2}{4 \times k} + h\right) + \left(\frac{4}{t_{clk}}(k + 1)\right) \quad (7)$$

позволяет определить оптимальное разбиение совокупного и наперед заданного объема ассоциативной памяти [10–12]. В случае, представленном на рис. 4, лучшее разбиение есть минимум аддитивной функции, который определяется значением k , обращающим производную функции в нуль: $n \times n = 600 \times 600$, $h = 200$, $t_{clk} = 4$, $k = 4$.

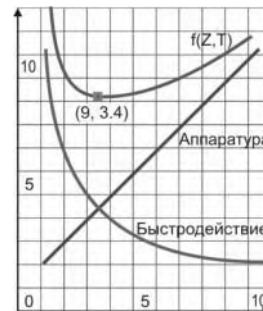


Рис. 4. Функции аппаратуры и времени от числа разбиений

Предложенная процесс-модель анализа графа ассоциативных таблиц, а также введенные критерии качества получаемых решений являются основой для разработки специализированной мультипроцессорной архитектуры, ориентированной на параллельное выполнение векторных логических операций.

3. АРХИТЕКТУРА ЛОГИЧЕСКОГО АССОЦИАТИВНОГО МУЛЬТИПРОЦЕССОРА

Для анализа больших информационных объемов логических данных существует несколько практически ориентированных технологий:

1. Использование рабочей станции для последовательного программирования задачи, где стоимость ее решения, а также временные затраты очень высоки.

2. Разработка специализированного параллельного процессора на основе ПЛИМ. Высокий параллелизм обработки информации компенсирует сравнительно низкую по сравнению с CPU тактовую частоту. Такое схемотехническое решение с возможностью перепрограммирования является по производительности выигрышным вариантом. Недостаток – отсутствие гибкости, характерной программным методам решения логических задач, и высокая стоимость реализации системы на кристалле ПЛИМ при больших объемах промышленного выпуска изделия.

3. Лучшее решение связано с объединением достоинств ЦПЭ, ПЛИМ и заказных СБИС [15, 16]. Это – гибкость программирования, возможность корректирования исходных кодов; минимальная мощность команд и простые схемотехнические решения аппаратной реализации мультипроцессора; распараллеливание логических процедур на структуре однобитовых процессоров. Имплементация мультипроцессора в кристалл заказной СБИС дает возможность получать максимальную тактовую частоту, минимальную стоимость микросхемы при больших объемах выпуска изделия, низкое энергопотребление.

Объединение преимуществ перечисленных технологий определяет базовую конфигурацию LAMP, который имеет сферическую структуру мультипроцессора (рис. 5), состоящую из 16 векторных секвенсоров (устройств последовательного управления), каждый из которых, включая граничные элементы, соединен с восемью соседними. Структура LAMP имеет прототип в виде процессора PRUS [17], разработанного доктором Stanley Hyduke (CEO Aldec, USA).

Занесение информации в процессор подобно классической схеме (design flow), за исключением того, что стадия размещения и трассировки заменяется фазой распределения программ и данных между всеми логическими бит-процессорами, работающими параллельно. Компилятор обеспечивает размещение данных по процессорам, задает время формирования решения на выходе каждого из них, а также планирует передачу полученных результатов другому процессору. LAMP есть эффективная сеть процессоров, которая обрабатывает данные и обеспечивает обмен информацией между компонентами сети в процессе их решения. Простая схемотехника каждого процессора позволяет эффективно обрабатывать сверхбольшие массивы, насчитывающие миллионы бит информации, затрачивая на это в сотни раз меньше времени по сравнению с универсальным процессором. Базовая ячейка – векторный процессор для LAMP – может быть синтезирован на 200 вентилях, что дает возможность сеть, содержащую 4096 вычислителей, легко реализовать в виде заказных СБИС, используя современную кремниевую технологию. Учитывая, что затраты памяти для хранения данных весьма незначительны, LAMP может представлять интерес для проектирования систем управления в таких областях человеческой деятельности, как индустрия, медицина, защита информации, геология, прогнозирование погоды, искусственный интеллект, космонавтика. LAMP представляет особый интерес для

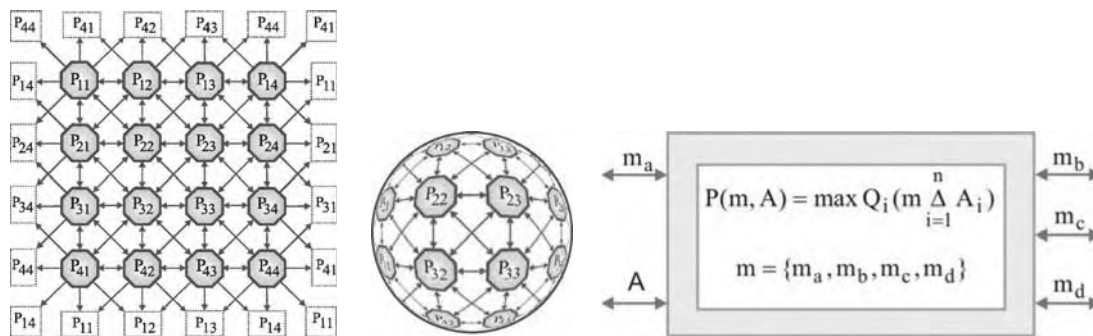


Рис. 5. Макроархитектура LAMP и интерфейс

цифровой обработки данных, распознавания образов и криптоанализа.

Если говорить о функционировании (системы) LAMP, то ее основная цель есть получение квазиоптимального решения в интегрированной задаче поиска и/или распознавания путем использования компонентов инфраструктуры, ориентированных на выполнение векторных логических операций: $P(m, A) = \min Q_i(m \Delta A_i)$, $m = \{m_a, m_b, m_c, m_d\}$. Интерфейс системы, соответствующий данному функционалу, представлен на рис. 5. Все компоненты $\{A, m_a, m_b, m_c, m_d\}$ могут быть как входными, так и выходными. Двухнаправленная детализация интерфейса связана с инвариантностью отношения всех переменных, векторов, A-матрицы и компонентов к входам и/или выходам инфраструктуры. Поэтому структурная модель системы LAMP может быть использована для решения любых задач прямой и обратной импликации в дискретном логическом пространстве, чем подчеркивается ее отличие от концепции автоматной модели вычислительного устройства с выраженными входами и выходами. Компоненты или регистры $m =$

(m_a, m_b, m_c, m_d) используются для получения решения в виде буферных, входных и выходных векторов, а также для идентификации оценки качества удовлетворения входного запроса.

Одним из возможных вариантов архитектуры мультипроцессора LAMP может служить структура, представленная на рис. 6. Основным компонентом структуры является мультипроцессорная матрица $P = [P_{ij}]$, $\text{card}(4 \times 4)$, содержащая 16 вектор-процессоров, каждый из которых предназначен для выполнения пяти логических векторных операций над содержимым памяти данных, представленной в виде таблицы, размерностью $A = \text{card}(m \times n)$.

Интерфейсный блок служит для обмена данными и загрузки программы обработки данных в соответствующую память команд. Блок управления осуществляет инициализацию выполнения команд логической обработки данных и синхронизирует функционирование всех компонентов мультипроцессора. Блок Infrastructure IP [1] предназначен для сервисного обслуживания всех модулей, диагностирования дефектов и восстановления работоспособности компонентов и устройства в целом. Элементарный

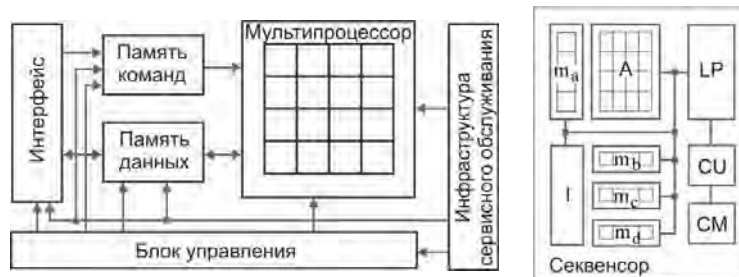


Рис. 6. Архитектура LAMP и структура секвенсора

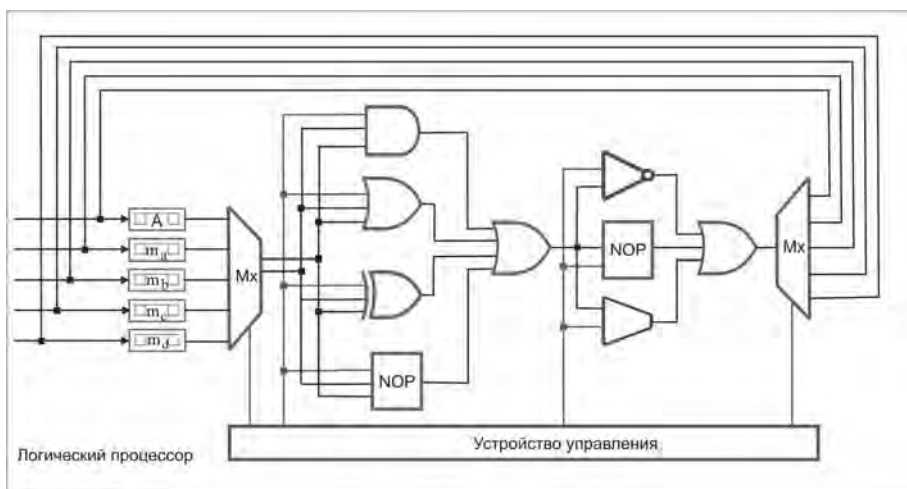


Рис. 7. Структура блока логических вычислений

логический ассоциативный процессор, или секвенсор (см. рис. 6), входящий в состав мультипроцессора, содержит: логический процессор (LP); ассоциативную (память) А-матрицу для параллельного выполнения базовых операций; блок векторов m , предназначенный для параллельного обслуживания строк и столбцов А-матрицы, а также обмена данными в процессе вычислений; память прямого доступа (СМ), сохраняющую команды программы обработки информации; автомат (СU) управления выполнением логических операций; интерфейс (I) связи секвенсора с другими элементами и устройствами мультипроцессора.

Логический процессор (LP) (рис. 7) осуществляет выполнение пяти операций: and, or, not, xor, slc (shift left bit crowding – уплотнение единиц со сдвигом влево), – которые являются базой для создания алгоритмов и процедур информационного поиска и оценивания решения.

Модуль LP имеет мультиплексор на входе для выбора двух из пяти операндов, которые подаются на выбранный логический векторный оператор. Сформированный результат через мультиплексор (элемент or) заносится в один из пяти операндов (память: вектор или матрица), который выбирается соответствующим адресом.

Особенности реализации логического процессора заключаются в наличии трех бинарных (and, or, xor) и двух унарных (not, slc) операций. Последние можно присоединять к такту обработки регистровых данных путем выбора одной из трех операций: not, slc, por (нет операции). Для повышения эффективности работы логического устройства вводятся два элемента с пустой операцией. Если, необходимо выполнить только унарную операцию, то на уровне бинарных команд следует выбрать por, что практически означает передачу данных через повторитель ко второму уровню унарных операций. Все операции в LP – регистровые или регистрово-матричные. Последние предназначены для анализа вектор-строк таблицы при использовании входного m -вектора как запроса для точного поиска информации. В блоке логических вычислений допустимо следующее сочетание операций и операндов:

$$C = \begin{cases} \{m_a, m_b, m_c, m_d\} \Delta A_i; \\ \{m_a, m_b, m_c, m_d\} \Delta \{m_a, m_b, m_c, m_d\}; \\ \{not, nop, slc\} \{m_a, m_b, m_c, m_d, A_i\}. \end{cases}$$

$$\Delta = \{and, or, xor\}.$$

Реализация всех векторных операций блока логических вычислений для одного секвенсора в среде

Verilog с последующей послесинтезной имплементацией в кристалл программируемой логики дает результаты:

Logic Block Utilization:

Number of 4 input LUTs: 400 out of 9,312 4%

Logic Distribution:

Number of occupied Slices: 200 out of 4,656 4%

Number of Slices only related logic: 200 out of 200 100 %

Total Number of 4 input LUTs: 400 out of 9,312 4%

Number of bonded IOBs: 88 out of 320 29%

Total equivalent gate count for design: 2400

Таковая частота выполнения регистровых операций в кристалле Virtex 4, Xilinx, равна 100 МГц, что существенно выше аналогичных процедур на компьютере с частотой 1 ГГц.

ВЫВОДЫ

Существующие программные аналоги не предлагают чисто ассоциативно-логических маршрутов поиска, распознавания и принятия решений в дискретном информационном пространстве [21, 22, 25]. Практически все они используют универсальную систему команд современного дорогостоящего процессора с математическим сопроцессором. С другой стороны, аппаратные специализированные средства логического анализа, которые могут выступать прототипами [4, 5], как правило, ориентированы на побитовую или не векторную обработку информации.

Для устранения недостатков программных аналогов и аппаратных прототипов в работе предложен новый подход векторно-логической обработки ассоциативных данных с полным исключением арифметических операций, влияющих на быстродействие и аппаратную сложность, который оказался удачно реализуемым на основе использования современной микроэлектронной аппаратуры в виде мультипроцессорной цифровой системы на кристалле.

Фактическая реализация подхода основывается на использовании новой инфраструктуры, которая включает следующие компоненты:

1. Процесс-модели анализа ассоциативных таблиц на основе использования векторных логических операций для поиска, распознавания образов, принятия и оценивания решений в векторном дискретном булевом пространстве. Модели ориентированы на достижение высокого быстродействия параллельного векторного логического анализа информации и подсчета критериев качества решения.

2. Мультипроцессорная архитектура параллельного решения ассоциативно-логических задач с минимальным множеством векторных логических операций и полным исключением арифметических ко-

манд, что обеспечивает высокое быстродействие, минимальную стоимость и незначительное энергопотребление LAMP, имплементированного в кристалл программируемой логики.

Достоверность и практическая значимость полученных результатов подтверждается созданием инфраструктуры мультипроцессора для диагностирования и восстановления работоспособности компонентов памяти в цифровой системе на кристалле. Дальнейшие исследования направлены на разработку прототипа логического ассоциативного мультипроцессора в целях решения актуальных задач поиска, распознавания и принятия решений с помощью предложенной инфраструктуры векторного логического анализа.

СПИСОК ЛІТЕРАТУРИ

- Zorian, Y. Test Strategies for System-in-Package / Y. Zorian // Plenary Paper of IEEE East-West Design & Test Symposium (EWDTS'08), Lviv, Ukraine. – Lviv, 2008.
- Smith, L. 3D Packaging Applications, Requirements, Infrastructure and Technologies / L. Smith // Fourth Annual International Wafer-Level Packaging Conference (San Jose, California, September, 2007). – San Jose, 2007.
- The next Step in Assembly and Packaging: System Level Integration in the package (SiP) / editors: William Chen, W. R. Bottoms, Klaus Pressel, Juergen Wolf // SiP White Paper. International Technology Roadmap for Semiconductors. – 2007. – P. 17–23.
- А. с. №1439682. 22.07.88. Регистр сдвига / Какурина Н. Я., Хаханов В. И., Лобода В. Г., Какурина А. Н. – 4 с.
- Бондаренко, М. Ф. О мозгоподобных ЭВМ / М. Ф. Бондаренко, З. В. Дударь, И. А. Ефимова, В. А. Лецинский, С. Ю. Шабанов-Кушнаренко // Радиоэлектроника и информатика. – 2004. – № 2. – С. 89–105.
- Бондаренко, М. Ф. Об алгебре предикатов / М. Ф. Бондаренко, Ю. П. Шабанов-Кушнаренко // Бионика интеллекта. – 2004. – № 1. – С. 15–26.
- Бондаренко, М. Ф. Теория интеллекта : учебник / М. Ф. Бондаренко, Ю. П. Шабанов-Кушнаренко. – Харьков : СМІТ, 2006. – 592 с.
- Бондаренко, М. Ф. Модели языка / М. Ф. Бондаренко, Ю. П. Шабанов-Кушнаренко // Бионика интеллекта. – 2004. – № 1. – С. 27–37.
- Акритас, А. Основы компьютерной алгебры с приложениями : пер. с англ. / А. Акритас. – М. : Мир, 1994. – 544 с.
- Гилл, Ф. Практическая оптимизация. / Ф. Гилл, У. Мюррей, М. Райт. – М. : Мир, 1985. – 509 с.
- Аттетков, А. В. Методы оптимизации / А. В. Аттетков, С. В. Галкин, В. С. Зарубин. – М. : Издательство МГТУ им. Н. Э. Баумана, 2003. – 440 с.
- Дегтярев, Ю. И. Методы оптимизации : учебное пособие для вузов / Ю. И. Дегтярев. – М. : Сов. радио, 1980. – 270 с.
- Bergeron, J. Writing Testbenches Using SystemVerilog / J. Bergeron // Springer Science and Business Media, 2006. – 414 p.
- Abramovici M. Digital System Testing and Testable Design / M. Abramovici, M.A. Breuer and A.D. Friedman. – Comp. Sc. Press, 1998. – 652 p.
- Densmore, D. A Platform-Based taxonomy for ESL Design / Douglas Densmore, Roberto Passerone, Alberto Sangiovanni-Vincentelli // Design & Test of computers. – 2006. – P. 359–373.
- Хаханов, В. И. Проектирование и тестирование цифровых систем на кристаллах / В. И. Хаханов, Е. И. Литвинова, О. А. Гузь. – Харьков : ХНУРЭ, 2009. – 484 с.
- Hahanov, V. I. SIGETEST – Test generation and fault simulation for digital design / V.I. Hahanov, D. M. Gorbunov, Y. V. Miroshnichenko, O. V. Melnikova, V. I. Obrikan, E. A. Kamenuka // Proc. of Conf. «Modern SoC Design Technology based on PLD». – Kharkov, 2003. – P. 50–53.
- Автоматизация диагностирования электронных устройств / Ю. В. Малышенко и др. ; под ред. В. П. Чипулиса. – М. : Энергоатомиздат, 1986. – 216 с.
- Хаханов, В. И. Проектирование и верификация цифровых систем на кристаллах / В. И. Хаханов, И. В. Хаханова, Е. И. Литвинова, О. А. Гузь. – Харьков : Новое слово, 2010. – 528 с.
- Cohen, A. A. Addressing architecture for Brain-like Massively Parallel Computers / A.A. Cohen // Euromicro Symposium on Digital System Design (DSD'04). – 2004. – P. 594–597.
- Липаев, В. В. Программная инженерия. Методологические основы : учебник / В. В. Липаев. – М. : Теис, 2006. – 608 с.
- Трахтенгерц, Э. А. Компьютерные методы реализации экономических и информационных управленческих решений / Э. А. Трахтенгерц. – М. : СИНТЕГ, 2009. – 396 с.
- Кузнецов, О. П. О моделировании быстрых интеллектуальных процессов обыденного мышления / О. П. Кузнецов // Интеллектуальные системы. – 1997. – Т. 2, вып. 1–4. – Стр. 52–57.
- Кузнецов, О. П. Быстрые процессы мозга и обработка образов / О. П. Кузнецов // Новости искусственного интеллекта. – 1998. – № 2.
- Васильев, С. Н. Интеллектуальное управление динамическими системами / С. Н. Васильев, А. К. Жерлов, Е. А. Федосов, Б. Е. Федун. – М. : Физ.-мат. лит.-ра, 2000. – 352 с.

Надійшла 18.06.2010

Хаханов В. И., Чумаченко С. В., Литвинова Е. И., Гузь О. А.
МУЛЬТИПРОЦЕССОР ДЛЯ АНАЛІЗУ ІНФОРМАЦІЙНОГО ПРОСТОРУ

Запропоновано архітектуру швидкодіючого мультипроцесора паралельного аналізу інформації, представленої у вигляді аналітичних, графових і табличних структур асоціативних відношень, для пошуку, розпізнавання та прийняття рішень у n -вимірному векторному дискретному просторі. Розглядаються векторно-логічні процес-моделі актуальних прикладних задач, якість рішення яких оцінюється введеною інтегральною неарифметичною метрикою взаємодії булевих векторів.

Ключові слова: мультипроцесор, аналіз інформації, логічне асоціативне відношення, процес-модель.

Hahanov V. I., Chumachenko S. V., Litvinova E. I., Guz O. A.
MULTIPROCESSOR FOR INFORMATION SPACE ANALYZING

Novel process-models for analyzing information in the tabular form based on using vector logical operations to solve the problems of search, diagnosis, pattern recognition and decision-making in the vector discrete Boolean space are proposed. The models are focused on realization of high-performance vector concurrent logical analysis of information that in the limit completely excludes the use of arithmetic operations.

Key words: multiprocessor, information analyzing, logical associative relationship, process-model.

ТЕОРИЯ И МЕТОДИ АВТОМАТИЧНОГО УПРАВЛІННЯ

ТЕОРИЯ И МЕТОДЫ АВТОМАТИЧЕСКОГО УПРАВЛЕНИЯ

THEORY AND METHODS OF AUTOMATIC CONTROL

УДК 62-55:681.515

Гостев В. И.

*Д-р техн. наук, заведующий кафедрой Государственного университета информационно-коммуникационных технологий,
г. Киев*

ПРОЕКТИРОВАНИЕ НЕЧЕТКОГО РЕГУЛЯТОРА ПРИ ИДЕНТИЧНЫХ ГАУССОВЫХ ФУНКЦИЯХ ПРИНАДЛЕЖНОСТИ

Получены аналитические выражения для управляющих воздействий на выходе нечеткого регулятора при идентичных гауссовых функциях принадлежности, изложены вопросы проектирования нечеткого регулятора и предложена практическая схема нечеткого регулятора.

Ключевые слова: автоматическое управление, нечеткий регулятор, MATLAB, нечеткая логика.

ВВЕДЕНИЕ

Метод проектирования нечетких регуляторов на основе пакета нечеткой логики системы MATLAB достаточно подробно изложен, например, в работах [1, 2]. В работе [3] предложен новый метод проектирования одного класса нечетких регуляторов, основанный на полученных аналитических выражениях для управляющих воздействий на выходе нечеткого регулятора при различных функциях принадлежности с двумя термами. Представлена функциональная схема нечеткого регулятора, на базе которой возможна реализация нечетких регуляторов программным или аппаратным способом. При проектировании нечетких регуляторов предложенным методом нет необходимости в использовании пакета нечеткой логики системы MATLAB, и процедура проектирования нечетких регуляторов упрощается. Нечеткий регулятор представляется в виде по-

следовательного соединения трех блоков (см. рис. 1): 1 – формирователя величин $A(t)$ и $B(t)$, 2 – блока сравнения величин $A(t)$ и $B(t)$ и расчета u_c (u_c – ненормированное управляющее воздействие на выходе нечеткого регулятора на оси универсального множества $U = [0, 1]$), 3 – блока нормировки выходной величины.

Ниже изложено проектирование нечеткого регулятора при идентичных гауссовых функциях принадлежности.

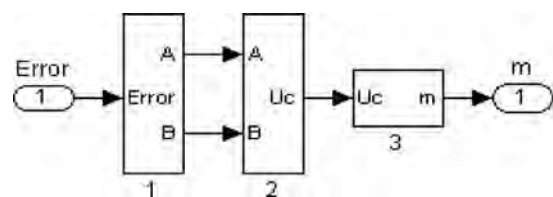


Рис. 1

**АНАЛИТИЧЕСКИЕ ВЫРАЖЕНИЯ
ДЛЯ УПРАВЛЯЮЩИХ ВОЗДЕЙСТВИЙ
НА ВЫХОДЕ НЕЧЕТКОГО РЕГУЛЯТОРА
ПРИ ИДЕНТИЧНЫХ ГАУССОВЫХ
ФУНКЦИЯХ ПРИНАДЛЕЖНОСТИ**

Рассмотрим нечеткий регулятор, на вход которого поступают ошибка системы θ , первая производная ошибки $\dot{\theta}$ и вторая производная ошибки $\ddot{\theta}$. Нечеткий регулятор практически реализуется на микроЭВМ (или микропроцессоре) и работает в дискретном режиме, поэтому на входе регулятора включается аналого-цифровой преобразователь АЦП, а на выходе – цифроаналоговый преобразователь ЦАП. АЦП квантует непрерывную ошибку системы управления $\theta(t) = u(t) - x(t)$ с шагом квантования h . В качестве первой и второй производных от ошибки вычисляем первую и вторую разность по формулам

$$\left. \begin{aligned} \dot{\theta}(k) &= [\theta(k) - \theta(k-1)]/h; \\ \ddot{\theta}(k) &= [\dot{\theta}(k) - \dot{\theta}(k-1)]/h = \\ &= [\theta(k) - 2\theta(k-1) + \theta(k-2)]/h^2, \end{aligned} \right\} \quad (1)$$

где $\theta(k)$ – квантованная ошибка на выходе АЦП. ЦАП представляет собой, как правило, фиксатор нулевого порядка с передаточной функцией $H(s) = (1 - e^{-hs})/s$.

Пусть на универсальном множестве $U = [0, 1]$ заданы два нечетких подмножества, функции принадлежности (ФП) которых для каждой лингвистической величины определяются по формулам (см. рис. 2):

$$\mu_1(u) = \exp\left(-\frac{u^2}{2c^2}\right), \quad u \in [0, 1];$$

$$\mu_2(u) = \exp\left[-\frac{(u-1)^2}{2c^2}\right], \quad u \in [0, 1].$$

При поступлении на нечеткий регулятор в какой-то фиксированный момент времени значений входных переменных θ^* , $\dot{\theta}^*$ и $\ddot{\theta}^*$ с шагом квантования h осуществляется пересчет входных переменных в пе-

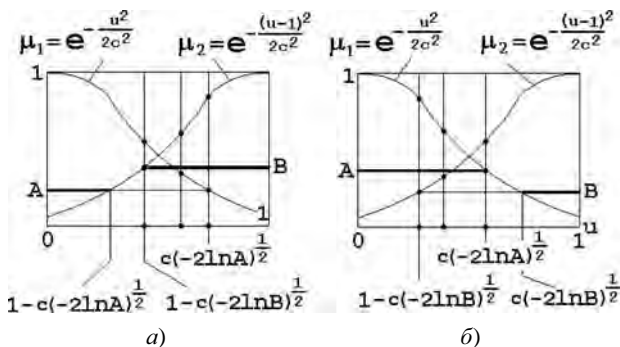


Рис. 2

ременные u_1^* , u_2^* , u_3^* на универсальном множестве $U = [0, 1]$ и расчет значений ФП для этих переменных (см. рис. 2). Точками на универсальном множестве отмечены возможные для какого-то момента времени значения переменных u_1^* , u_2^* , u_3^* .

Для упрощения нормировки (пересчета значений сигналов в значения элементов единого универсального множества) диапазоны изменения входных сигналов (параметров нечеткого регулятора) принимаем симметричными:

$$A_m = \theta_{\max} = -\theta_{\min}, \quad B_m = \dot{\theta}_{\max} = -\dot{\theta}_{\min},$$

$$C_m = \ddot{\theta}_{\max} = -\ddot{\theta}_{\min}.$$

Тогда формулы для нормировки (пересчета) принимают вид

$$\left. \begin{aligned} u_1^* &= (\theta^* + A_m)/(2A_m); \\ u_2^* &= (\dot{\theta}^* + B_m)/(2B_m); \\ u_3^* &= (\ddot{\theta}^* + C_m)/(2C_m). \end{aligned} \right\} \quad (2)$$

Лингвистическое правило управления нечеткого регулятора формулируется в виде

$$\text{Если } (\theta^* = a_1^j) \text{ и } (\dot{\theta}^* = a_2^j) \text{ и } (\ddot{\theta}^* = a_3^j),$$

$$\text{то } (m^* = a_c^j), \quad j = \overline{1, 2}, \quad (3)$$

где a_1^j , a_2^j и a_3^j – лингвистические оценки ошибки, первой производной ошибки и второй производной ошибки, рассматриваемые как нечеткие терм-множества, определенные на универсальном множестве, $j = \overline{1, 2}$; a_c^j – лингвистические оценки управляющего воздействия на объект, выбираемые из терм-множества переменной m . Лингвистические оценки выбираются из терм-множеств лингвистических переменных θ^* , $\dot{\theta}^*$, $\ddot{\theta}^*$ и m^* :

$$a_c^j \in \{\text{отрицательная (1), положительная (2)}\}.$$

В соответствии с лингвистическими правилами управления функция принадлежности управляющего воздействия $\mu_c^1(u)$ нечеткому множеству «отрицательный» ограничена сверху значением

$$A = \min[\mu_1(u_1^*), \mu_1(u_2^*), \mu_1(u_3^*)]; \quad (4)$$

функция принадлежности управляющего воздействия $\mu_c^2(u)$ нечеткому множеству «положительный» ограничена сверху значением

$$B = \min[\mu_2(u_1^*), \mu_2(u_2^*), \mu_2(u_3^*)]. \quad (5)$$

Результирующая функция принадлежности для управляющего воздействия получается формированием максимума

$$\mu_c(u) = \max[\mu_c^1(u), \mu_c^2(u)]. \quad (6)$$

Для определения конкретного значения управляющего воздействия m^* формируется «результующая фигура», ограниченная результирующей ФП, и производится поиск абсциссы «центра тяжести результирующей фигуры» u_c .

Какие бы значения ни принимали переменные u_1^* , u_2^* , u_3^* на универсальном множестве $U = [0, 1]$ в зависимости от соотношений величин A и B «результующая фигура» может принимать только две конфигурации: при $A \leq B$ первая конфигурация показана

на рис. 2, а; при $A \geq B$ вторая конфигурация показана на рис. 2, б.

Общая формула для определения абсциссы «центра тяжести результирующей фигуры» записывается в виде

$$u_c = \int_0^1 u \mu(u) du / \int_0^1 \mu(u) du. \quad (7)$$

Абсцисса «центра тяжести результирующей фигуры» при $A \leq B$ определяется по формуле

$$u_c = \frac{\frac{A}{2} - cA\sqrt{-2\ln A} + cB\sqrt{-2\ln B} + c\sqrt{\frac{\pi}{2}}[\operatorname{erf}(\sqrt{-\ln A}) - \operatorname{erf}(\sqrt{-\ln B})] - c^2(B - A + A\ln A - B\ln B)}{A - cA\sqrt{-2\ln A} + cB\sqrt{-2\ln B} + c\sqrt{\frac{\pi}{2}}[\operatorname{erf}(\sqrt{-\ln A}) - \operatorname{erf}(\sqrt{-\ln B})]} \text{ при } A \leq B. \quad (8)$$

Абсцисса «центра тяжести результирующей фигуры» при $A \geq B$ определяется по формуле

$$u_c = \frac{\frac{B}{2} - c^2(B - A + A\ln A - B\ln B)}{B + cA\sqrt{-2\ln A} - cB\sqrt{-2\ln B} + c\sqrt{\frac{\pi}{2}}[\operatorname{erf}(\sqrt{-\ln B}) - \operatorname{erf}(\sqrt{-\ln A})]} \text{ при } A \geq B. \quad (9)$$

Полученное значение u_c затем преобразуется в значение управляющего воздействия на объект управления (при симметричном диапазоне изменения выходного сигнала $D_m = m_{\max} = -m_{\min}$):

$$m^* = m_{\min}(1 - 2u_c) = 2D_m u_c - D_m. \quad (10)$$

В качестве примера приведем следующие результаты расчетов.

При $A = 0,02$, $B = 0,4$, $c = 0,3 \Rightarrow u_c = 0,695$. При $A = 0,4$, $B = 0,02$, $c = 0,3 \Rightarrow u_c = 0,305$.

ПРИНЦИПАЛЬНАЯ СХЕМА НЕЧЕТКОГО РЕГУЛЯТОРА С ИДЕНТИЧНЫМИ ГАУССОВЫМИ ФУНКЦИЯМИ ПРИНАДЛЕЖНОСТИ

Выполненная с использованием интерактивной системы MATLAB принципиальная схема нечеткого регулятора с идентичными гауссовыми функциями принадлежности представлена на рис. 3.

Формирователь величин $A(t)$ и $B(t)$ реализован на основе формул (1), (2), (4) и (5). Элементами ограничения (Saturation) моделируем универсальное множество $U = [0, 1]$, на которое поступают переменные u_i , $i = 1, 2, 3$. В блоках Fcn, Fcn1, Fcn2 записываем аналитические выражения для функций принадлежности $\mu_1(u)$, а в блоках Fcn3, Fcn4, Fcn5 – аналитические выражения для функций принадлежности $\mu_2(u)$. На выходе блоков Fcn, Fcn1, Fcn2 получаем переменные $\mu_2(u_i)$ (соответственно $\mu_1(u_1)$, $\mu_1(u_2)$,

$\mu_1(u_3)$), а на выходе блоков Fcn3, Fcn4, Fcn5 получаем переменные $\mu_2(u_i)$ (соответственно $\mu_2(u_1)$, $\mu_2(u_2)$, $\mu_2(u_3)$).

Блок сравнения величин $A(t)$ и $B(t)$ и расчета u_c (отдельно показан на рис. 4) реализован на основе формул (8) и (9). На выходе верхнего сумматора формируется числитель, а на выходе нижнего сумматора формируется знаменатель выражения (8) и на выходе делителя Product4 формируется величина u_c при $A \leq B$. Аналогичным способом на выходе верхнего сумматора формируется числитель, а на выходе нижнего сумматора формируется знаменатель выражения (9) и на выходе делителя Product5 формируется величина u_c при $A \geq B$. Переключатель Switch замыкает верхний контакт при условии $A \leq B$ (когда на среднем контакте сигнал положительный, в блоке Switch параметр Threshold = 0,000001). При условии $A \geq B$, когда на среднем контакте переключателя Switch сигнал отрицательный, переключатель замыкает нижний контакт. На выходе переключателя Switch получаем значение u_c .

В средстве моделирования и исследования систем управления с обратной связью Simulink функция erf в блоке User-Defined Functions моделируется не блоком Fcn, а блоком MATLAB Fcn, поэтому блок сравнения величин A и B и расчета u_c , показанный на рис. 3, представлен моделью с блоками MATLAB Fcn1 – MATLAB Fcn8, в которых записа-

ны те же выражения, что и в соответствующих блоках на рис. 4.

Функции принадлежности в формирователе величин $A(t)$ и $B(t)$ называются *входными*, а в блоке сравнения

величин $A(t)$ и $B(t)$ и расчета u_c называются *выходными*. Для данного регулятора эти функции идентичны.

Блок нормировки выходной переменной реализован на основе формулы (10).

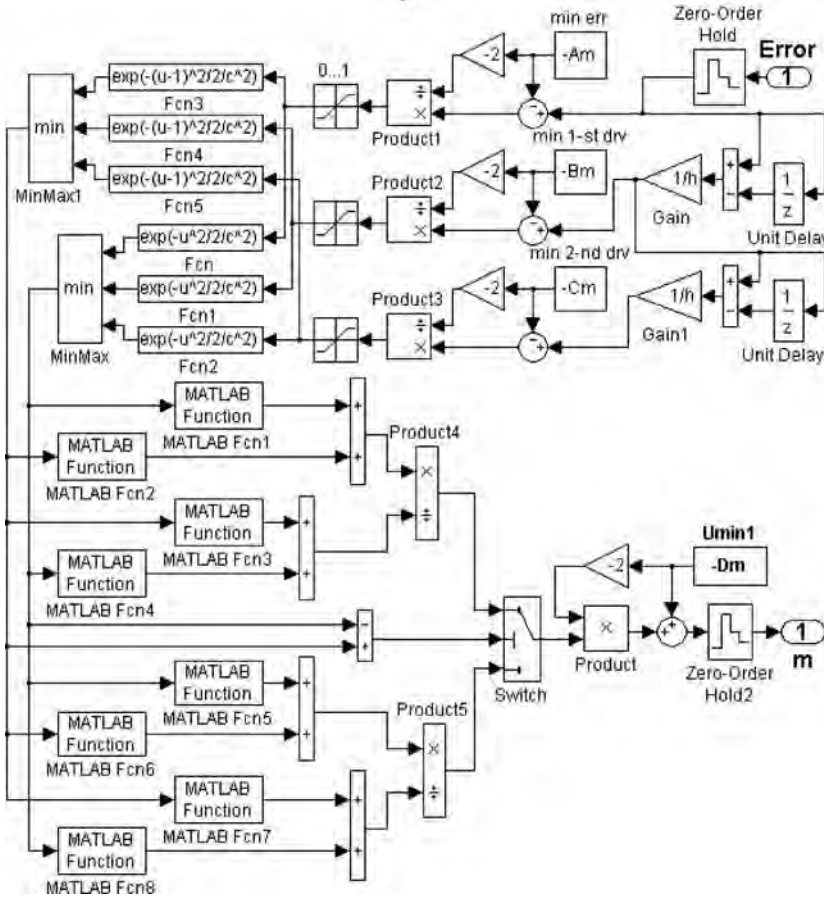


Рис. 3

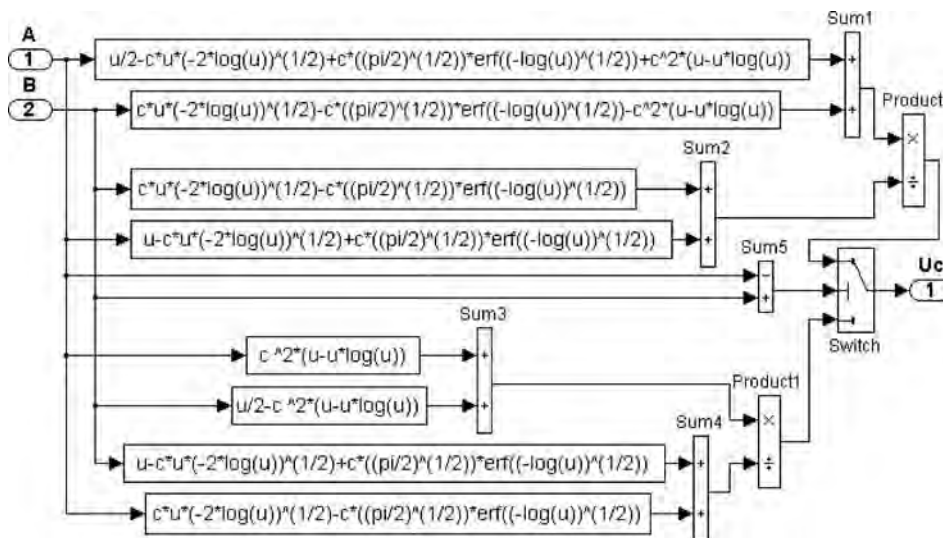


Рис. 4

ЛОГИКА РАБОТЫ НЕЧЕТКОГО РЕГУЛЯТОРА

Логика работы нечеткого регулятора (см. рис. 3) для фиксированного момента времени отображена на рис. 5.

В формирователе величин $A(t)$ и $B(t)$ на входе нечеткого регулятора ошибка системы (Error) θ^* квантуется АЦП (Zero-Older Hold), вычисляются первая (1-st drv) $\dot{\theta}^*$ и вторая (2-nd drv) $\ddot{\theta}^*$ разности от ошибки с шагом квантования h , и значения входных переменных θ^* , $\dot{\theta}^*$ и $\ddot{\theta}^*$ пересчитываются в переменные u_1^* , u_2^* , u_3^* по формулам (2). Производится расчет значений входных функций принадлежности (см. рис. 5, а, б)

$$\mu_1(u) = \exp\left(-\frac{u^2}{2c^2}\right), u \in [0, 1];$$

$$\mu_2(u) = \exp\left[-\frac{(u-1)^2}{2c^2}\right], u \in [0, 1].$$

для переменных u_1^* , u_2^* , u_3^* и на выходе блоков Min-Max и MinMax1 на основе алгоритма Мамданиопределяются соответственно величины A и B по формулам (4), (5).

Величины A и B поступают на блок сравнения величин $A(t)$ и $B(t)$ и расчета u_c , в котором производится расчет ненормированного выхода регулятора по фор-

мулам (8) и (9) для выходных функций принадлежности, которые идентичны входным (см. рис. 5, в, з). Далее полученное значение u_c в блоке нормировки выходной переменной пересчитывается в выходное напряжение регулятора по формуле (10).

В динамике на выходе блоков Product, Product1, Product2 структурной схемы формирователя величин $A(t)$ и $B(t)$ получаем переменные u_i (соответственно u_1, u_2, u_3). Выражения (4) и (5) на каждом шаге h вычисляются в блоках MinMax и MinMax1, на выходе которых получают значения переменных $A(t)$ и $B(t)$, и в блоке сравнения по формулам (8) и (9) вычисляются значения переменной $u_c(t)$, которые преобразуется блоком нормировки выходной переменной в значение управляющего воздействия на объект управления $m(t)$.

В схеме формирователя величин $A(t)$ и $B(t)$ при настройке нечеткого регулятора перестраиваются граничные значения диапазонов $A_m = \theta_{\max} = -\theta_{\min}$, $B_m = \dot{\theta}_{\max} = -\dot{\theta}_{\min}$, $C_m = \ddot{\theta}_{\max} = -\ddot{\theta}_{\min}$. В блоке нормировки выходной переменной перестраиваются граничные значения диапазона $D_m = m_{\max} = -m_{\min}$.

ВЫВОД

Изложенные теория и практическая схема нечеткого регулятора с идентичными гауссовыми функциями принадлежности дают возможность использовать такой регулятор в различных системах автоматического управления и путем настройки параметров регулятора добиваться высокого качества систем.

СПИСОК ЛИТЕРАТУРЫ

1. Дьяконов, В. Математические пакеты расширения MATLAB. Специальный справочник / Дьяконов В., Круглов В. – СПб. : Питер, 2001. – 480 с.
2. Гостев, В. И. Синтез нечетких регуляторов систем автоматического управления / Гостев В. И. – К. : Радиоаматор, 2005. – 708 с.
3. Гостев, В. И. Новый метод проектирования одного класса нечетких цифровых регуляторов / Гостев В. И. // Проблемы управления и информатики. – 2007. – № 6. – С. 73–84.

Надійшла 16.09.2010

Гостев В. И.

ПРОЕКТУВАННЯ НЕЧІТКОГО РЕГУЛЯТОРА ПРИ ІДЕНТИЧНИХ ГАУССОВИХ ФУНКЦІЯХ ПРИНАЛЕЖНОСТІ

Отримано аналітичні вирази для керуючих впливів на виході нечіткого регулятора при ідентичних гауссових функціях приналежності, викладено питання проектування нечіткого регулятора та запропоновано практичну схему нечіткого регулятора.

Ключові слова: автоматичне керування, нечіткий регулятор, MATLAB, нечітка логіка.

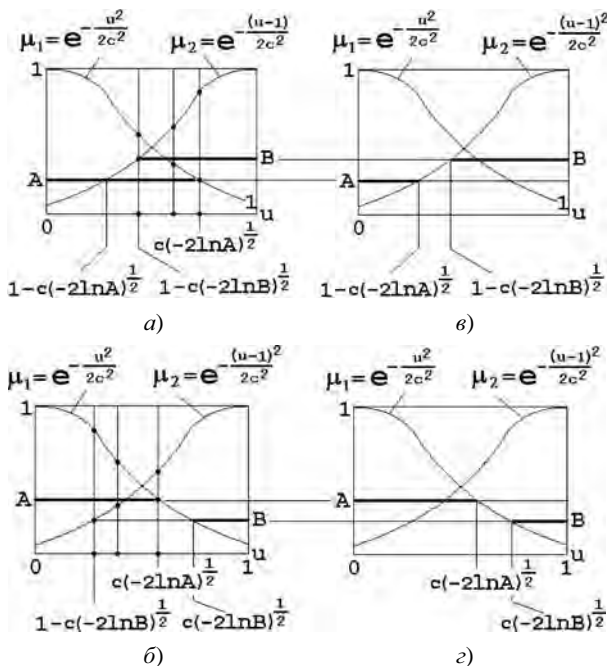


Рис. 5

Gostev V. I.

DESIGNING OF A FUZZY CONTROLLER AT IDENTICAL GAUSS MEMBERSHIP FUNCTIONS

Analytical expressions for control actions at the fuzzy controller output at identical gauss membership functions have

been obtained, fuzzy controller designing procedure is described, and the practical scheme of a fuzzy controller is proposed.

Key words: automatic control, fuzzy controller, MATLAB, fuzzy logic.

УДК 62-50

Кудин В. Ф.¹, Колесниченко С. П.²¹Д-р техн. наук, профессор Национального технического университета Украины «Киевский политехнический институт»²Канд. техн. наук, старший преподаватель Национального технического университета Украины «Киевский политехнический институт»

СУБОПТИМАЛЬНОЕ НЕЛИНЕЙНОЕ УПРАВЛЕНИЕ ПО КРИТЕРИЮ БЫСТРОДЕЙСТВИЯ НА ОСНОВЕ МЕТОДА БЕЛЛМАНА – ЛЯПУНОВА

Рассматривается общетеоретическая задача синтеза субоптимального нелинейного управления на примере двухмассовой электромеханической системы управления крановым механизмом передвижения с учетом гашения колебаний транспортного груза. Задача решается на базе метода Беллмана – Ляпунова с использованием концепции «инвариантного погружения» по критерию быстродействия. Проведено исследование динамики замкнутой системы с синтезированным субоптимальным регулятором.

Ключевые слова: электромеханическая система, аналитическое конструирование регуляторов, критерий быстродействия, нелинейное управление, метод Беллмана – Ляпунова.

ВВЕДЕНИЕ

Задачи синтеза алгоритмов управления, оптимальных по быстродействию, остаются актуальными для проектирования автоматических систем управления различными транспортными механизмами производственных цехов, следящих систем различного назначения, систем управления колебаниями, манипуляторами и др. [1–4]. Известно, что в практике проектирования систем максимального быстродействия синтез алгоритмов выполняется, как правило, методом фазовой плоскости. Возможности применения этого метода ограничены объектами третьего порядка, передаточные функции которых не имеют комплексно-сопряженных полюсов, а фазовые траектории являются монотонными кривыми.

В настоящее время уделяется большое внимание проблеме синтеза оптимального управления двухмассовой электромеханической системой (ЭМС) механизма перемещения крана с гашением колебаний подвешенного груза [1]. При этом рассматривается довольно широкий спектр математических моделей управляемой ЭМС, которая учитывает нелинейность объекта управления и электромагнитную инерционность электропривода. Математические модели подобных управляемых систем наряду с аperiodическими звеньями зачастую содержат и колебательные. Синтез оптимальных по быстродействию управлений такими системами, как показывают выполненные в

[5, 6] исследования, представляют трудную, возможно, даже неразрешимую задачу.

В целом исследуемая модель ЭМС является нелинейной по переменным состояния и управляющему воздействию. Кроме того, минимизируемые функционалы, используемые при построении оптимальных двухмассовых ЭМС, являются, обычно, неквадратичными, что усложняет процедуру аналитического конструирования оптимальных регуляторов (АКОР).

В данной статье предлагается приближенное решение нелинейной задачи АКОР по быстродействию на основе метода Беллмана – Ляпунова в сочетании с концепцией «инвариантного погружения» [7–10]. Метод обладает вычислительной эффективностью и легко распространяется на нелинейные системы высокой размерности.

1. ПОСТАНОВКА ЗАДАЧИ

Задана математическая модель колебаний подвешенного груза в виде консервативного звена с электроприводом, создающим динамический момент (рис. 1) [3].

На рис. 1 используются следующие обозначения: m_1 – масса тележки (моста); m_2 – масса груза; φ – угол отклонения груза от вертикали; F – динамическое усилие, приложенное к тележке (мосту); M – динамический момент электропривода; ρ – радиус приведения.

© Кудин В. Ф., Колесниченко С. П., 2011

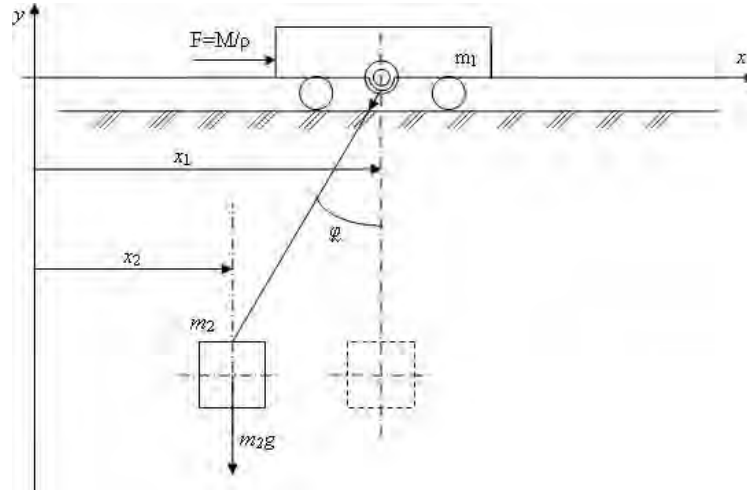


Рис. 1. Подъемно-транспортный механизм с маятниковой подвеской груза

При малых отклонениях груза от вертикали движение двухмассовой системы «тележка – груз» описывается системой линейных уравнений [3]

$$\frac{d^2\varphi}{dt^2} + \omega_0^2\varphi = b_1F,$$

$$\frac{dF}{dt} = -\frac{1}{T_a}F + b_2u, [u] \leq \bar{u},$$

где $\omega_0 = 1,33 \text{ c}^{-1}$ – частота собственных колебаний системы «тележка – груз»; $T_a = 0,05 \text{ c}$ – электромагнитная постоянная момента электропривода; $b_1 = 6,54 \times 10^{-4}$, $b_2 = 7,6 \times 10^4$ – коэффициенты; u – управление.

Запишем заданную систему дифференциальных уравнений в форме Коши:

$$\begin{aligned} \dot{x}_1 &= x_2 & x_1 - \varphi \\ \dot{x}_2 &= -\omega_0^2 x_1 + b_1 x_3, \text{ здесь } x_2 - \dot{\varphi} \\ \dot{x}_3 &= -a_{31} x_3 + b_2 u & x_3 - F, \quad u = f(\sigma) = \sigma^{1/5}. \end{aligned} \quad (1)$$

Примечание. Учет ограничения на управление u с помощью некоторой функции $u = f(\sigma)$ был впервые предложен А. Миеле [11]. Здесь роль управляющего воздействия переходит уже к σ , на которое не наложено ограничение. Поэтому формировать подынтегральную функцию $w(\sigma)$, которая определяет вид $u = f(\sigma)$, гораздо легче. При этом зависимость вида $u = \sqrt[2n-2]{\sigma}$ ($n = 2, 3, 4, \dots$) позволяет при достаточно больших n аппроксимировать релейную характеристику с большой точностью на классе гладких функций.

Ставится задача отыскания управления $\sigma = \gamma(x_1, x_2, x_3)$, обеспечивающего переход изображающей точки пространства состояний в начало координат

при произвольных начальных условиях, на решениях системы (1), исходя из минимизации функционала вида

$$\min_{\sigma} J = \int_0^{\infty} (W(x_1, x_2, x_3) + c\sigma^2) dt. \quad (2)$$

Рассмотрим сначала процедуру АКОР, учитывая ограничение на управление, используя нелинейное преобразование А. Миеле, не уточняя вид функции $W(x_1, x_2, x_3)$.

Метод динамического программирования Беллмана дает следующее функциональное уравнение:

$$\begin{aligned} \min_{\sigma} \left[W(x_1, x_2, x_3) + c\sigma^2 + \frac{\partial V}{\partial x_1} x_2 + \right. \\ \left. + \frac{\partial V}{\partial x_2} (-\omega_0^2 x_1 + b_1 x_3) + \right. \\ \left. + \frac{\partial V}{\partial x_3} (-a_{31} x_3 + b_2 \sigma^{1/5}) \right] = 0. \end{aligned} \quad (3)$$

После дифференцирования получаем

$$2c\sigma + \frac{b}{5} \frac{\partial V}{\partial x_3} \sigma^{-4/5} = 0; \quad \sigma^{9/5} + \frac{b_2}{10c} \frac{\partial V}{\partial x_3} = 0;$$

$$\sigma = -\left(\frac{b}{10c} \frac{\partial V}{\partial x_3}\right)^{5/9}; \quad u = -\sqrt[9]{\frac{b_2}{10c} \frac{\partial V}{\partial x_3}}.$$

Исключая σ из функционального уравнения (3), получаем уравнение Гамильтона – Якоби – Беллмана (ГЯБ) вида

$$\begin{aligned} W(x_1, x_2, x_3) + \frac{\partial V}{\partial x_1} x_2 + \frac{\partial V}{\partial x_2} (-\omega_0^2 x_1 + b_1 x_3) + \\ + \frac{\partial V}{\partial x_3} (-a_{31} x_3) = 9c \left(\frac{b_2}{10c} \frac{\partial V}{\partial x_3}\right)^{10/9}. \end{aligned} \quad (4)$$

Разлагая правую часть в цепную дробь [12], получаем

$$W(x_1, x_2, x_3) + \frac{\partial V}{\partial x_1}x_2 + \frac{\partial V}{\partial x_2}(-\omega_0^2x_1 + b_1x_3) - \frac{\partial V}{\partial x_3}ax_3 = \frac{\mu_{11}\sigma_1^2 + \mu_{12}\sigma_1^4 + \mu_{13}\sigma_1^6}{1 + \mu_{21}\sigma^2 + \mu_{22}\sigma_1^4}, \quad (5)$$

где $\sigma_1 = \frac{b_2}{6c} \frac{\partial V}{\partial x_3}$, $\mu_{11}, \mu_{12}, \dots$ – постоянные коэффициенты разложения.

Приближенное решение этого уравнения находится в виде последовательности степенных форм от переменных состояния.

$$V(x) = v^2(x) + v^4(x) + v^6(x) \dots \quad (6)$$

Здесь $v^2(x)$ – квадратичная форма от переменных x_1, x_2, x_3 , $v^4(x)$ – форма четвертой степени и т. д. Параметры квадратичной формы определяются системой уравнений Риккати, а параметры четвертичной формы – системой линейных алгебраических уравнений.

В точках позиционирования тележки перемещаемый груз занимает произвольное положение. Возникает типичная задача АКОР по отработке начальных условий, исходя из условия быстрейшего затухания переходного процесса [или критерия $\min_t \max_t x(t)$]. В этом случае критерий оптимальности и подынтегральные функции приобретают следующий вид:

$$\min_{|u| \leq \bar{u}} J = \int_0^T [W(x_1, x_2, x_3) + cu^2] dt, \quad (7)$$

где $W_1(x_1, x_2, x_3) = 1$.

Уравнение ГЯБ для функционала быстродействия с учетом ограничения на управление будет иметь следующий вид при $u = f(\sigma) = \sigma^{1/5}$:

$$1 + \frac{\partial V}{\partial x_1}x_2 + \frac{\partial V}{\partial x_2}(-\omega_0^2x_1 + b_1x_3) - \frac{\partial V}{\partial x_3}ax_3 = 9c \left(\frac{b_2}{10c} \frac{\partial V}{\partial x_3} \right)^{10/9}. \quad (8)$$

По утверждению М. Атанса, П. Фалба, решение задачи синтеза по критерию быстродействия на базе метода динамического программирования, которое сводится к решению уравнения ГЯБ, практически невозможно.

Один из возможных путей – переход к аналитическому выражению подынтегральной функции критерия быстродействия в функции переменных состояния.

Подынтегральную функцию $W_1(x_1, x_2, x_3) = 1$ можно аппроксимировать соотношением $W_1(x_1, x_2, x_3) =$

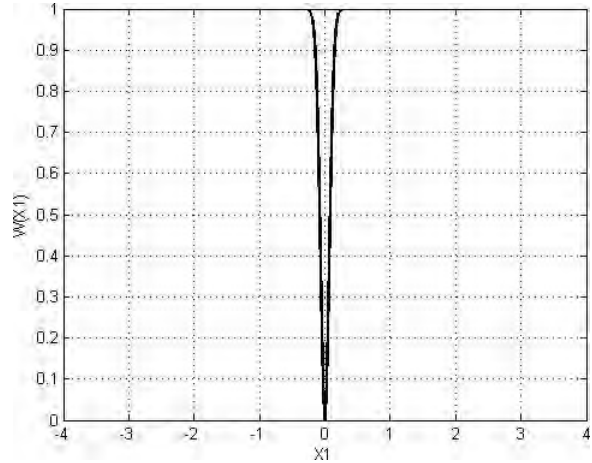


Рис. 2. Аппроксимированная подынтегральная функция $W_1(x_1)$

$= 1 \approx 1 - e^{-\alpha x_1^2}$. Данная функция при $a = 100$ представлена на рис. 2.

Аппроксимированная подынтегральная функция уже является аналитической. Поэтому решение задачи синтеза (уравнение ГЯБ) можно искать в виде многомерного степенного ряда. Тогда для системы дифференциальных уравнений (1) и функционала (7) уравнение ГЯБ будет иметь вид

$$\frac{(\alpha x)^2 + 1/2(\alpha x)^4 + 1/6(\alpha x)^6}{1 + (\alpha x)^2 + 1/2(\alpha x)^4 + 1/6(\alpha x)^6} + \frac{\partial V}{\partial x_1}x_2 + \frac{\partial V}{\partial x_2}(-\omega_0^2x_1 + b_1x_3) - \frac{\partial V}{\partial x_3}ax_3 = 9c \left(\frac{b_2}{10c} \frac{\partial V}{\partial x_3} \right)^{10/9}. \quad (9)$$

После разложения правой части уравнения в цепную дробь решение уравнения ГЯБ ищется в виде последовательности степенных форм. Однако процедура решения уравнения ГЯБ достаточно сложна в вычислительном отношении. Поэтому используем процедуру синтеза субоптимального управления на базе метода Беллмана – Ляпунова в сочетании с принципом «инвариантного погружения» [6–9].

2. РЕШЕНИЕ ЗАДАЧИ АКОР

Преобразуем подынтегральную функцию минимизируемого функционала

$$W_1(x_1, x_2) = 1 = \alpha(x_1)x_1^2,$$

где $\alpha(x_1) = \frac{1}{x_1^2}$ – весовая константа.

В итоге получаем критерий оптимальности вида

$$\min_{\sigma} I = \int_0^{\infty} (\alpha(x_1)x_1^2 + c\sigma^2) dt, \quad (10)$$

т. е. неквадратичный функционал сведен к квадратичному функционалу, весовая константа которого $\alpha(x_1)$ является функцией переменной x_1 . Тогда для системы дифференциальных уравнений (1) и функционала (10) получаем уравнения ГЯБ в частных производных для фиксированной области фазового пространства.

$$\alpha(x_1)x_1^2 + \frac{\partial V}{\partial x_1}x_2 + \frac{\partial V}{\partial x_2}(-\omega_0^2x_1 + b_1x_3) - \frac{\partial V}{\partial x_3}ax_3 = \frac{q^2(a)b_2^2}{4c} \left(\frac{\partial V}{\partial x_3} \right)^2. \quad (11)$$

Здесь $q(a)$ есть постоянная величина для фиксированной области фазового пространства.

Трансформация уравнения ГЯБ (8) в (11) обусловлена тем, что управляющее воздействие есть $u = f(\sigma)$, где $f(\sigma)$ – функция нелинейного преобразования А. Miele, представленная в виде $u = q(a)\sigma = \frac{f(\sigma)}{\sigma}\sigma$. Здесь $q(a)$ – коэффициент гармонической линеаризации для затухающих переходных процессов [13, 14]. Этот коэффициент является приближенным эквивалентом коэффициента линеаризации по методу секущей, поскольку $q(a)$ также можно получить методом среднеквадратичного приближения [15].

Система уравнений Риккати имеет вид

$$\begin{aligned} \alpha(x_1) - 2k_{21}\omega_0^2 &= \frac{q^2(a)b_2^2}{c}k_{31}^2, \\ 2k_{12} &= \frac{q^2(a)b_2^2}{c}k_{23}^2, \\ 2b_1k_{13} &= \frac{q^2(a)b_2^2}{c}k_{33}^2, \\ k_{13} - b_1k_{12} - a_{31}k_{32} &= \frac{q^2(a)b_2^2}{c}k_{31}k_{32}, \\ k_{13} - bk_{12} - a_{31}k_{32} &= \frac{q^2(a)b_2^2}{c}k_{32}k_{33}, \\ k_{12}b_1 - \omega_0^2k_{23} - a_{31}k_{31} &= \frac{q^2(a)b_2^2}{c}k_{13}k_{33}. \end{aligned} \quad (12)$$

Далее осуществляем решение задачи АКОР «в малом» при $\alpha(x_1) = \alpha_1(x_1) = 400$, $q(a) = q_1(a) = 6$, $c = c_1 = 1$.

На этом этапе получаем закон управления

$$\begin{aligned} \sigma_1(x) &= -\frac{q_1(x)}{c_1}(k_{31}x_1 + k_{32}x_2 + k_{33}x_3) = \\ &= -(k_1x_1 + k_2x_2 + k_3x_3) = \\ &= -(19,75x_1 + 2,382x_2 + 0,00x_3). \end{aligned}$$

На втором этапе получаем решение задачи АКОР «в большом» при $\alpha(x_1) = \alpha_2(x_1) = 4$, $q(a) = q_2(a) = 0,8$, $c = c_2 = 1/(50)^2$:

$$\begin{aligned} \sigma_2(x) &= -\frac{q_2(x)}{c_2}(k_{31}x_1 + k_{32}x_2 + k_{33}x_3) = \\ &= -(k_1'x_1 + k_2'x_2 + k_3'x_3) = \\ &= -(98,28x_1 + 13,88x_2 + 0,000x_3). \end{aligned}$$

Таким образом, получена последовательность «мгновенных» оптимальных управлений для некоторых совокупностей начальных условий. Далее возникает необходимость сшивания «мгновенных значений» оптимального управления u_1 и u_2 . Находим вариации параметров Δk_i , возникающих при переходе из одной области в другую, и рассматриваем их как управляющие воздействия.

Тогда

$$u_2 = -\text{sign}\sigma_2 = -\text{sign}[k_1x_1 + k_2x_2 + k_3x_3 + \Delta k_1x_1 + \Delta k_2x_2 + \Delta k_3x_3], \quad (13)$$

где Δk_1 , Δk_2 и Δk_3 определяем из соотношения

$$\begin{aligned} \Delta\sigma(x) &= \sigma_2(x) - \sigma_1(x) = \\ &= (k_1' - k_1)x_1 + (k_2' - k_2)x_2 + (k_3' - k_3)x_3 = \\ &= \Delta k_1x_1 + \Delta k_2x_2 + \Delta k_3x_3. \end{aligned}$$

Дальнейшая процедура синтеза производится в соответствии с методикой, изложенной в [9, 10]. Ставится задача минимизации функционала

$$\min_{\Delta k_1, \Delta k_2} J = \int_0^{\infty} \left(\sum_{i=1}^{n=3} \alpha_i x_i^2 + c_1 \Delta k_1^2 + c_2 \Delta k_2^2 \right) dt \quad (14)$$

на решениях системы дифференциальных уравнений

$$\begin{aligned} \dot{x}_1 &= x_2, \\ \dot{x}_2 &= -\omega_0^2 x_1 + b_1 x_3, \\ \dot{x}_3 &= -a_{31} x_3 + b_2 u, \\ u &= -f(\sigma) = -\text{sign} \left[\sum_{i=1}^3 k_i x_i + \Delta k_1 x_1 + \Delta k_2 x_2 \right]. \end{aligned} \quad (15)$$

Процесс определения допустимого управления по переменной состояния x_3 приводит к ничтожно малому значению эффективности управления, что позволяет в (14) и (15), соответственно, пренебречь слагаемыми $c_3 \Delta k_3^2$ и $\Delta k_3 x_3$.

Получаем функциональное уравнение Беллмана для системы (15) и функционала (14):

$$\min_{\Delta k_1, \Delta k_2} \left[\sum_{i=1}^{n=3} \alpha_i x_i^2 + c_1 \Delta k_1^2 + c_2 \Delta k_2^2 + \frac{\partial V}{\partial x_1} x_2 + \frac{\partial V}{\partial x_2} (-\omega_0^2 x_1 + a_i x_3) + \frac{\partial V}{\partial x_3} (-a_{31} x^3 + b_2 q_3(a)) \times \left(\sum_{i=1}^{n=3} k_i x_i + \Delta k_1 x_1 + \Delta k_2 x_2 \right) \right] = 0. \quad (16)$$

После дифференцирования получаем

$$\Delta k_1 = -\frac{b_2 q_2(a)}{c_1} \frac{\partial V}{\partial x_3} x_1; \quad \Delta k_2 = -\frac{b_2 q_2(a)}{c_2} \frac{\partial V}{\partial x_3} x_2. \quad (17)$$

Подставляя (17) в (16), получаем в итоге модифицированное уравнение Гамильтона – Якоби – Беллмана вида

$$\sum_{i=1}^{n=3} \alpha_i x_i^2 + \frac{\partial V}{\partial x_1} x_2 + \frac{\partial V}{\partial x_2} (-\omega_0^2 x_1 - k_\phi x_3) - \frac{\partial V}{\partial x_3} (a_{31} x_3 + b_2 q_2(a)) \sum_{i=1}^{n=3} k_i x_i = \sum_{i=1}^{n=2} \frac{x_i^2}{4c_2} \left[b_2 q_2(a) \frac{\partial V}{\partial x_3} \right]^2. \quad (18)$$

Решение этого уравнения аппроксимируется последовательностью степенных форм

$$V(x) = \sum_{q=1}^{\infty} v^{2q}(x) = v^2(x) + v^4(x) + v^6(x) \dots \quad (19)$$

Параметры квадратичной и последующих форм определяются из системы линейных алгебраических уравнений. Окончательно получаем субоптимальный закон управления:

$$u = -\text{sign}[k_1 x_1 + k_2 x_2 + k_3 x_3 + q_2(a) \sum_{i=1}^{n=2} \frac{x_i}{2c} \frac{\partial V}{\partial x_3}].$$

Используя только квадратичную форму последовательности степенных форм (10), получаем нелинейный закон управления:

$$u = -\text{sign}[k_1 x_1 + k_2 x_2 + k_3 x_3 + (k_{21} x_1^2 + k_{22} x_2^2) \times (k_{31} x_1 + k_{32} x_2 + k_{33} x_3)]. \quad (20)$$

Исследование динамики замкнутого контура с синтезированным регулятором произведено методом цифрового моделирования. На рис. 3 представлены переходные процессы изменения угла ϕ и управления u при отработке начального отклонения в 0,5 рад и ограничении на управление.

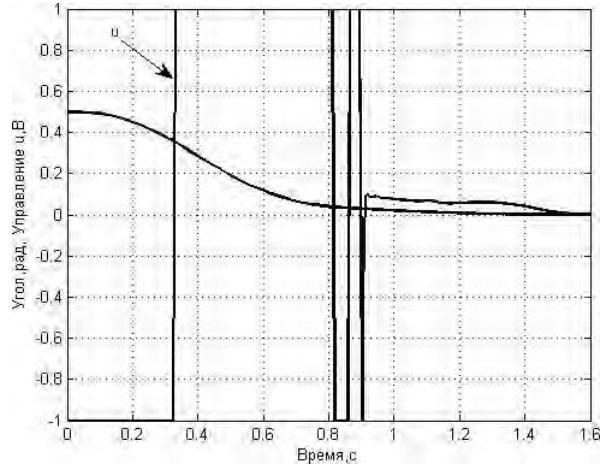


Рис. 3. Результаты моделирования

ЗАКЛЮЧЕНИЕ

Таким образом, исходная нелинейная задача АКОР для функционала (7) и системы (1) свелась к решению задачи минимизации квадратичного функционала (10) на решении линеаризованной системы (1), т. е. к решению ряда линейных задач АКОР. Дальнейшие исследования по синтезу систем управления желательно осуществить для нелинейной модели транспортируемого груза по фазовым координатам при ограничении управления. Предложенная методика позволяет существенно усилить вычислительную эффективность процедуры АКОР, что крайне важно при решении прикладных задач высокой размерности.

СПИСОК ЛИТЕРАТУРЫ

1. Динамика машин и управление машинами / под ред. Крейнина Г. В. – М. : Машиностроение, 1988. – 240 с.
2. Терехов, В. М. Системы управления электроприводов / Терехов, В. М., Осипов О. И. – М. : Академия, 2005. – 240 с.
3. Герасимьяк, Р. П. Электроприводы крановых механизмов / Герасимьяк Р. П., Параил В. А. – М. : Энергия, 1970. – 136 с.
4. Герасимьяк, Р. П. Анализ и синтез крановых электромеханических систем / Герасимьяк Р. П., Лещев В. А. – Одесса: СНИЛ, 2008. – 192 с.
5. Федун, Б. Е. Синтез оптимального по быстродействию управления колебательным звеном / Б. Е. Федун // Известия РАН. Теория и системы управления. – 2000. – № 3. – С. 78–84.
6. Крутько, П. Д. Исследование динамики субоптимальных по быстродействию автоматических систем / П. Д. Крутько // Известия РАН. Теория и системы управления. – 2004. – № 2. – С. 16–33.
7. Беллман, Р. Методы вычислений : избранные главы // Автоматика и телемеханика. – 1993. – № 8. – С. 3–39 ; № 9. – С. 3–51; № 10. – С. 3–43.
8. Булатов, В. Н. Методы погружения в задачах оптимизации / Булатов В. Н. – Новосибирск : Наука, 1977. – 154 с.
9. Кудин, В. Ф. Аналитическое конструирование нелинейных регуляторов с помощью метода гармоничес-

- кой линеаризации // Электромеханика. Известия ВУЗов. – 1989. – № 9. – С. 60–66.
10. *Kudin, V.* Synthesis of suboptimal nonlinear regulator by immersion method / *Kudin V., Kolachny I.* // *Jour. Electrical engineering.* – 1998. – Vol. 49, № 1–2. – Pp. 11–15.
 11. *Miele, A.* General variational theory of the flight paths of rocket-powered aircraft, missiles and satellite carriers / *Miele A.* // *Astronaut Acta.* – 1958. – Vol.4. – Pp. 264–288.
 12. *Хованский, А. Н.* Приложение цепных дробей и их обобщений к вопросам приближенного анализа / *Хованский А.Н.* – М. : Гостехиздат, 1956.– 203 с.
 13. *Попов, Е. П.* Прикладная теория процессов управления в нелинейных системах / *Попов Е. П.* – М. : Наука, 1973. – 584 с.
 14. *Пальтов, И. П.* Качество процессов и синтез корректирующих устройств в нелинейных автоматических системах / *Пальтов И. П.* – М. : Наука, 1975. – 367 с.
 15. *Кринецкий, И. И.* Расчет нелинейных автоматических систем / *Кринецкий И. И.* – Киев : Техника, 1968. – 312 с.

Надійшла 07.10.2010

Кудін В. Ф., Колесніченко С. П.

СУБОПТИМАЛЬНЕ НЕЛІНІЙНЕ КЕРУВАННЯ ЗА КРИТЕРІЄМ ШВИДКОДІЇ НА ОСНОВІ МЕТОДУ БЕЛЛМАНА – ЛЯПУНОВА

Розглядається загальнотеоретична задача синтезу субоптимального нелінійного керування на прикладі двомасової електромеханічної системи керування крановим механізмом пересування з урахуванням гасіння коливаний

транспортного вантажу. Задача вирішується на базі методу Беллмана – Ляпунова з використанням концепції «інваріантного занурення» за критерієм швидкодії. Проведено дослідження динаміки замкнутої системи із синтезованим субоптимальним регулятором.

Ключові слова: електромеханічна система, аналітичне конструювання регуляторів, критерій швидкодії, нелінійне керування, метод Беллмана – Ляпунова.

Kudin V. F., Kolesnichenko S. P.

SUBOPTIMUM NONLINEAR CONTROL BY OPERATION SPEED CRITERION BASED ON BELLMAN-LYAPUNOV METHOD

A general-theoretical task of suboptimal nonlinear control algorithm synthesis is discussed. As an example, a two-mass electromechanical control system of a crane moving mechanism is examined taking into account load oscillations damping. The task is solved on the basis of Bellman–Lyapunov method using the concept of «invariant immersion» by operation speed criterion. The dynamics of a closed-loop system with a synthesized suboptimal regulator is investigated.

Key words: electromechanical system, analytical regulator synthesis, operation speed criterion, nonlinear control, Bellman–Lyapunov method.

УДК 629.424.2

Орловский И. А.

Канд. техн. наук, доцент Запорожского национального технического университета

ИДЕНТИФИКАЦИЯ МАТЕМАТИЧЕСКОЙ МОДЕЛИ В ВИДЕ ПОЛИНОМИАЛЬНОЙ РЕКУРРЕНТНОЙ НЕЙРОННОЙ СЕТИ И НАСТРОЙКА РЕГУЛЯТОРОВ ЭЛЕКТРОПРИВОДА С СЕРИЕСНЫМ ДВИГАТЕЛЕМ

Синтезированы математические модели электропривода с серийным двигателем в виде полиномиальных рекуррентных нейронных сетей (ПРНС) по данным режима его работы. Исследованы способы идентификации параметров привода и зависимости момента сопротивления от скорости двигателя при задании специальных режимов работы и разных видов уравнений, описывающих нелинейности параметров. Выполнена настройка ПИ-регулятора скорости на полученной модели в виде ПРНС.

Ключевые слова: математическая модель, двигатель постоянного тока последовательного возбуждения, рекуррентная нейронная сеть, регулятор скорости, идентификация параметров.

ВВЕДЕНИЕ

В настоящее время имеется возможность применить в электроприводе (ЭП) с микропроцессорной системой управления (СУ) методы синтеза математических моделей электромеханических систем в виде нейронных сетей [1, 2] с последующей оптимальной настройкой регуляторов на этих моделях [3, 4]. Поиск с помощью различных алгоритмов оптимальных параметров регуляторов с помощью математических

моделей предъявляют повышенные требования к точности моделей и их обобщающим свойствам. В работах [5, 6] разработан метод синтеза математических моделей нелинейных электромеханических объектов в виде полиномиальных рекуррентных нейронных сетей (ПРНС), однако отсутствует методика идентификации с высокой точностью с помощью полученных моделей параметров объекта (как линейных, так и нелинейных). В работе [7]

исследована возможность идентификации параметров двигателя постоянного тока последовательного возбуждения (ДПТТВ) по его обученным моделям в виде ПРНС. При этом на обучающих наборах обеспечивалась высокая точность всей модели, однако точность идентификации нескольких параметров ЭП была низкой. Такой двигатель является основным элементом тягового ЭП, идентификация параметров которого и настройка СУ ЭП в реальном времени являются актуальными и сложными задачами.

Математическая модель ЭП в виде ПРНС, полученная по данным режима его работы при разных видах нелинейностей параметров, имеет различную точность. В научной литературе отсутствуют исследования настройки регуляторов ЭП с помощью таких моделей.

Цель статьи – повышение точности идентификации нелинейных параметров ДПТТВ по его рассчитанным по данным режима работы математическим моделям в виде ПРНС, а также настройка регуляторов на полученных моделях.

ПОСТАНОВКА ЗАДАЧИ ИССЛЕДОВАНИЯ

Согласно предложенному в работе [5] методу, структура математической модели нелинейных объектов определяется по известной информации о структуре их математической модели и характере имеющихся нелинейностей параметров. Поэтому для синтеза математической модели в виде ПРНС воспользуемся следующим описанием процессов в ДПТТВ с нелинейной зависимостью момента сопротивления от скорости [6, 7]:

$$\left. \begin{aligned} L(I) \frac{dI}{dt} + I \cdot R_d &= U - c\Phi(I) \cdot \omega, \\ J \frac{d\omega}{dt} &= c\Phi(I) \cdot I - M_c(\omega), \end{aligned} \right\} \quad (1)$$

где R_d – эквивалентное активное сопротивление цепи постоянного тока; $L(I)$ – эквивалентная индуктивность цепи постоянного тока, зависящая от тока якоря двигателя; $c\Phi(I)$ – произведение конструктивной постоянной «с» двигателя на значение магнитного потока Φ , зависящего от тока якоря; J – суммарный момент инерции на валу двигателя; $M_c(\omega)$ – момент сопротивления, зависящий от угловой скорости двигателя ω .

Для данного объекта вектором состояния является $x = [I, \omega]^T$; вектором входных сигналов – $u = [U, 1]^T$. Второй элемент вектора u взят равным единице для обеспечения общности выражений при описании момента сопротивления в виде нелинейно-

го коэффициента $M_c(\omega)$. С учетом сделанных обозначений система (1) может быть представлена в матричном виде:

$$\dot{x} = Ax + Bu, \quad (2)$$

где матрицы нелинейных коэффициентов A и B имеют вид

$$\left. \begin{aligned} A &= \begin{bmatrix} -R_d L^{-1}(I) & -L^{-1}(I)c\Phi(I) \\ J^{-1}c\Phi(I) & 0 \end{bmatrix}, \\ B &= \begin{bmatrix} L^{-1}(I) & 0 \\ 0 & -J^{-1}M_c(\omega) \end{bmatrix}. \end{aligned} \right\} \quad (3)$$

Для получения общих выражений расчета весовых коэффициентов ПРНС (с использованием которых формируются полиномы имеющихся в объекте нелинейностей) запишем систему уравнений (2) в виде

$$\dot{x} = CY, \quad (4)$$

где $Y = [I, \omega, U_d, 1]^T$ – вектор, объединяющий векторы состояния объекта и входных сигналов. Матрица C объединяет матрицы A и B .

Структурная схема модели ДПТТВ в виде ПРНС с использованием полиномиальных блоков [5, 7] дана на рис. 1. Полиномиальные блоки PO L формируют произведения (с единичными коэффициентами) по-

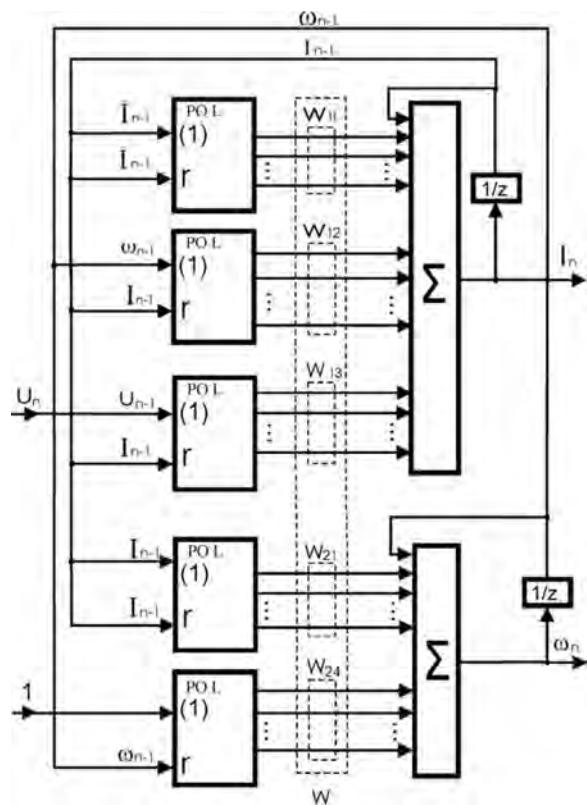


Рис. 1. Структура модели ДПТТВ в виде ПРНС

линомиальных членов степени r от сигналов, поступающих на входы « r », на сигналы, поступающие на входы «(1)». Выходы блоков POL для первого и второго нейронов, соответственно, обозначим векторами h_j и f_j с такими же индексами j , как у элементов векторов весовых коэффициентов w_{ij} и элементов c_{ij} матрицы C . Для дискретной системы с тактом счета T при аппроксимации нелинейностей полиномами второго порядка вектора h и f , соответственно, для первого и второго нейронов, в n -м такте определяются уравнениями

$$\left. \begin{aligned} h_n &= [I_{n-1}, I_{n-1}^2, I_{n-1}^3, \omega_{n-1}, \omega_{n-1} I_{n-1}, \\ &\quad \omega_{n-1} I_{n-1}^2, U_n, U_n I_{n-1}, U_n I_{n-1}^2]^T, \\ f_n &= [I_{n-1}, I_{n-1}^2, I_{n-1}^3, 1, \omega_{n-1}, \omega_{n-1}^2]^T. \end{aligned} \right\} \quad (5)$$

С учетом приведенных выше обозначений, системе (2) представим в разностном виде:

$$\left. \begin{aligned} I_n &= I_{n-1} + w_{11} h_{1n} + w_{12} h_{2n} + w_{13} h_{3n}, \\ \omega_n &= \omega_{n-1} + w_{21} f_{1n} + w_{14} f_{4n}, \end{aligned} \right\} \quad (6)$$

или более компактно следующим образом:

$$\left. \begin{aligned} w_1 h_n &= \Delta I_n, \\ w_2 f_n &= \Delta \omega_n, \end{aligned} \right\} \quad (7)$$

где $w_1 = [w_{11}, w_{12}, w_{13}]$, $w_2 = [w_{21}, w_{24}]$, $h = [h_1, h_2, h_3]^T$, $f = [f_1, f_4]^T$, $\Delta I_n = I_n - I_{n-1}$, $\Delta \omega_n = \omega_n - \omega_{n-1}$.

При выполнении M измерений входных сигналов и вектора состояния объекта каждое из уравнений системы (7) дает M уравнений:

$$w_1^* h^* = \Delta I_n^*, w_2^* f^* = \Delta \omega_n^*,$$

где $w_1^* = [w_1, \dots, w_1]^T$, $w_2^* = [w_2, \dots, w_2]^T$ – векторы, каждый размером $(M \times 1)$, $h^* = [h_n, \dots, h_{n-M+1}]$, $f^* = [f_n, \dots, f_{n-M+1}]$, $\Delta I_n^* = [\Delta I_n, \dots, \Delta I_{n-M+1}]^T$, $\Delta \omega_n^* = [\Delta \omega_n, \dots, \Delta \omega_{n-M+1}]^T$. Определение весовых коэффициентов выполняется путем расчета минимального среднеквадратичного отклонения для всех уравнений с помощью псевдообратных матриц $(h^*)^+$ и $(f^*)^+$ по выражениям [6]

$$\left. \begin{aligned} w_1 &= \Delta I_n^* (h^*)^+, \\ w_2 &= \Delta \omega_n^* (f^*)^+. \end{aligned} \right\} \quad (8)$$

Весовые коэффициенты ПРНС, исходя из уравнения (4), могут быть вычислены по математическим моделям ДПТПВ по формуле [7]

$$W = \begin{bmatrix} w_{11} & w_{12} & w_{13} & 0 \\ w_{21} & 0 & 0 & w_{24} \end{bmatrix} = \text{Pol}_r(TC), \quad (9)$$

где $w_{ij} = \text{Pol}_r(Tc_{ij})$; $\text{Pol}_r(\)$ – функция, определяющая вектор коэффициентов полинома степени r для выражения, находящегося в скобках. При этом

$$\left. \begin{aligned} w_{11} &= T\text{Pol}(-R_d L^{-1}(I)), \\ w_{12} &= T\text{Pol}(-L^{-1}(I)c\Phi(I)), \\ w_{13} &= T\text{Pol}(L^{-1}(I)), \\ w_{21} &= T\text{Pol}(J^{-1}c\Phi(I)), \\ w_{24} &= T\text{Pol}(-J^{-1}M_c(\omega)). \end{aligned} \right\} \quad (10)$$

При расчете модели по данным режима работы одновременно находятся все весовые коэффициенты нейрона (выход нейрона формируется с учетом нескольких элементов матрицы C), обеспечивающие минимум среднеквадратичной ошибки его выходного сигнала для конкретного набора данных режима. При этом в общем случае возникают ошибки идентификации параметров объекта по его модели в виде ПРНС, вызванные тем, что критерий точной идентификации каждого отдельного нелинейного элемента матрицы C не ставится.

Общее решение задачи одновременной идентификации нескольких параметров нелинейных объектов в научной литературе в настоящее время отсутствует. Поэтому исследовались три случая идентификации для нескольких специальных режимов электропривода и для нелинейностей заданного вида, а именно:

1. Нелинейности в пределах изменения переменных режима работы точно описываются полиномами заданной степени.
2. Использование дополнительной информации о математических преобразованиях, учитывающих структуру модели объекта. С этой целью выполняются дополнительные функциональные преобразования того же набора измеряемых данных, например дифференцирование дифференциального уравнения. Используется также задание режимов, в которых мгновенные значения части координат объекта повторяются при разных мгновенных значениях остальной части координат.
3. Задание режимов, когда часть элементов вектора состояния равна нулю, например, остановка двигателя на свободном выбеге, разрыв якорной цепи двигателя (ток якоря равен нулю); угловая скорость двигателя равна нулю (фиксация ротора двигателя при пониженном напряжении).

Для первого случая, когда нелинейность точно задана полиномами степени r , значения полиномиаль-

ных коэффициентов обратного значения индуктивности находятся из третьего уравнения системы (10):

$$\begin{aligned} L_0^{-1} &= T^{-1} \cdot w_{130}, L_1^{-1} = T^{-1} \cdot w_{131}, \dots, \\ L_r^{-1} &= T^{-1} \cdot w_{13r}. \end{aligned} \quad (11)$$

При известном моменте инерции полиномиальные коэффициенты момента сопротивления и магнитного потока определяются по следующим зависимостям:

$$\left. \begin{aligned} M_{c0} &= -T^{-1} \cdot J \cdot w_{240}, \\ M_{c1} &= -T^{-1} \cdot J \cdot w_{241}, \\ \dots \\ M_{cr} &= -T^{-1} \cdot J \cdot w_{24r}, \\ c\Phi_0 &= T^{-1} \cdot J \cdot w_{210}, \\ c\Phi_1 &= T^{-1} \cdot J \cdot w_{211}, \\ \dots \\ c\Phi_r &= T^{-1} \cdot J \cdot w_{21r}, \end{aligned} \right\} \quad (12)$$

где w_{ijk} – весовые коэффициенты ПРНС, формирующие коэффициенты полиномиального разложения нелинейностей.

В общем случае произведение $c\Phi$ может быть вычислено делением полинома из второго уравнения системы (10) на полином из третьего уравнения. Сопротивление R_d находится делением полиномов первого уравнения системы (10) на полином третьего уравнения. Суммарный момент инерции на валу ДППВ находится делением полинома $c\Phi(I)$ на полином $w_{21}(I)$. Для исследования первого случая нелинейности магнитного потока от тока $c\Phi(I)$ и момента сопротивления от угловой скорости $M_c(\omega)$ задавались в виде полиномов второй степени $c\Phi = c\Phi_0 + c\Phi_1 I + c\Phi_2 I^2$ и $M_c = M_{c0} + M_{c1} \omega + M_{c2} \omega^2$.

Во втором случае для сближения критериев, обеспечивающих точный выход нейрона и точное формирование отдельных нелинейных параметров, введена дополнительная информация при расчете ПРНС в виде математической модели вычисления производных измеряемых сигналов. Для вывода уравнений расчета весовых коэффициентов проделаем следующее.

Запишем разностные уравнения (7) через элементы c_{11} , c_{12} и c_{13} матрицы C в виде степенных зависимостей

$$\left. \begin{aligned} (I_n - I_{n-1})/T &= (c_{110} + c_{111} \cdot I_{n-1}^1 + c_{112} \cdot I_{n-1}^2) \cdot I_{n-1} + \\ &+ (c_{120} + c_{121} \cdot I_{n-1}^1 + c_{122} \cdot I_{n-1}^2) \cdot \omega_{n-1} + \\ &+ (c_{130} + c_{131} \cdot I_{n-1}^1 + c_{132} \cdot I_{n-1}^2) \cdot U_n, \\ (\omega_n - \omega_{n-1})/T &= (c_{210} + c_{211} I_{n-1}^1 + c_{212} I_{n-1}^2) I_{n-1} + \\ &+ (c_{240} + c_{241} \omega_{n-1}^1 + c_{242} \omega_{n-1}^2). \end{aligned} \right\} \quad (13)$$

После дифференцирования уравнений системы (13) и обозначения вторых разностей $\delta I_n = I_n - 2I_{n-1} + I_{n-2}$ и $\delta \omega_n = \omega_n - 2\omega_{n-1} + \omega_{n-2}$ получим

$$\left. \begin{aligned} \delta I_n/T &= c_{110}(I_{n-1} - I_{n-2}) + c_{111}(2I_{n-1}^2 - 2I_{n-1} \cdot I_{n-2}) + \\ &+ c_{112}(3I_{n-1}^3 - 3I_{n-1}^2 \cdot I_{n-2}) + c_{120}(\omega_{n-1} - \omega_{n-2}) + \\ &+ c_{121}(2I_{n-1} \cdot \omega_{n-1} - I_{n-2} \cdot \omega_{n-1} - I_{n-1} \cdot \omega_{n-2}) + \\ &+ c_{122}(3I_{n-1}^2 \omega_{n-1} - 2I_{n-1} \cdot I_{n-2} \cdot \omega_{n-1} - I_{n-1}^2 \cdot \omega_{n-2}) + \\ &+ c_{130}(U_n - U_{n-1}) + c_{131}(2I_{n-1} \cdot U_n - I_{n-2} \cdot U_n - I_{n-1} \cdot U_{n-1}) + \\ &+ c_{132}(3I_{n-1}^2 \cdot U_n - 2I_{n-1} \cdot I_{n-2} \cdot U_n - I_{n-1}^2 \cdot U_{n-1}), \\ \delta \omega_n/T &= c_{210}(I_{n-1} - I_{n-2}) + \\ &+ c_{211}(I_{n-1}^2 - I_{n-1} \cdot I_{n-2} + I_{n-1}^2 - I_{n-1} \cdot I_{n-2}) + \\ &+ c_{212}(2I_{n-1}^3 - 2I_{n-1}^2 \cdot I_{n-2} + I_{n-1}^3 - I_{n-1}^2 \cdot I_{n-2}) + \\ &+ c_{241}(\omega_{n-1} - \omega_{n-2}) + c_{242}(2\omega_{n-1}^2 - 2\omega_{n-1} \cdot \omega_{n-2}). \end{aligned} \right\} \quad (14)$$

Обозначим векторами H_n и F_n сигналы функционального разложения при дифференцировании на входах первого и второго нейронов соответственно:

$$\begin{aligned} H_n &= \begin{bmatrix} I_{n-1} - I_{n-2} \\ 2I_{n-1}^2 - 2I_{n-1} \cdot I_{n-2} \\ 3I_{n-1}^3 - 3I_{n-1}^2 \cdot I_{n-2} \\ \omega_{n-1} - \omega_{n-2} \\ 2I_{n-1} \cdot \omega_{n-1} - I_{n-2} \cdot \omega_{n-1} - I_{n-1} \cdot \omega_{n-2} \\ 3I_{n-1}^2 \cdot \omega_{n-1} - 2I_{n-1} \cdot I_{n-2} \cdot \omega_{n-1} - I_{n-1}^2 \cdot \omega_{n-2} \\ U_n - U_{n-1} \\ 2I_{n-1} \cdot U_n - I_{n-2} \cdot U_n - I_{n-1} \cdot U_{n-1} \\ 3I_{n-1}^2 \cdot U_n - 2I_{n-1} \cdot I_{n-2} \cdot U_n - I_{n-1}^2 \cdot U_{n-1} \end{bmatrix}, \\ F_n &= \begin{bmatrix} I_{n-1} - I_{n-2} \\ 2I_{n-1}^2 - 2I_{n-1} \cdot I_{n-2} \\ 3I_{n-1}^3 - 3I_{n-1}^2 \cdot I_{n-2} \\ 0 \\ \omega_{n-1} - \omega_{n-2} \\ 2\omega_{n-1}^2 - 2\omega_{n-1} \cdot \omega_{n-2} \end{bmatrix}. \end{aligned} \quad (15)$$

Весовые коэффициенты $w_1 = [w_{110}, w_{111}, w_{112}, w_{120}, w_{121}, w_{122}, w_{130}, w_{131}, w_{132}]$ и $w_2 = [w_{210}, w_{211}, w_{212}, w_{240}, w_{241}, w_{242}]$ для моделей с дополнительным и без дополнительного дифференцирования одинаковые, тогда уравнения (13) и (14) можно записать в виде

$$\left. \begin{aligned} \Delta I_n &= w_1 \cdot h_n, \\ \delta I_n &= w_1 \cdot H_n, \\ \Delta \omega_n &= w_2 \cdot f_n, \\ \delta \omega_n &= w_2 \cdot F_n. \end{aligned} \right\} \quad (16)$$

При использовании M измерений вектора $\Delta I, \delta I, h, H$ и $\Delta\omega, \delta\omega, f, F$ получают дополнительную размерность и примут вид

$$\left. \begin{aligned} \Delta I &= (\Delta I_n, \Delta I_{n-1}, \dots, \Delta I_{n-M+1})^T, \\ \delta I &= (\delta I_n, \delta I_{n-1}, \dots, \delta I_{n-M+1})^T, \\ h &= [h_n, h_{n-1}, \dots, h_{n-M+1}]^T, \\ H &= [H_n, H_{n-1}, \dots, H_{n-M+1}]^T, \\ \Delta\omega &= (\Delta\omega_n, \Delta\omega_{n-1}, \dots, \Delta\omega_{n-M+1})^T, \\ \delta\omega &= (\delta\omega_n, \delta\omega_{n-1}, \dots, \delta\omega_{n-M+1})^T, \\ f &= [f_n, f_{n-1}, \dots, f_{n-M+1}]^T, \\ F &= [F_n, F_{n-1}, \dots, F_{n-M+1}]^T. \end{aligned} \right\} \quad (17)$$

Для выполнения расчета весовых коэффициентов, соответствующих общему решению уравнений системы (16), обозначим для первого нейрона через DI вектор, объединяющий ΔI и δI , и через DH вектор, объединяющий h и H , аналогично для второго нейрона $D\omega$ – вектор, объединяющий $\Delta\omega$ и $\delta\omega$ и DF – вектор, объединяющий f и F :

$$\left. \begin{aligned} DI &= [\Delta I_n, \Delta I_{n-1}, \dots, \Delta I_{n-M+1}, \delta I_n, \delta I_{n-1}, \dots, \delta I_{n-M+1}]^T, \\ DH &= [h_n, h_{n-1}, \dots, h_{n-M+1}, H_n, H_{n-1}, \dots, H_{n-M+1}]^T, \\ D\omega &= [\Delta\omega_n, \Delta\omega_{n-1}, \dots, \Delta\omega_{n-M+1}, \delta\omega_n, \delta\omega_{n-1}, \dots, \delta\omega_{n-M+1}]^T, \\ DF &= [f_n, f_{n-1}, \dots, f_{n-M+1}, F_n, F_{n-1}, \dots, F_{n-M+1}]^T. \end{aligned} \right\} \quad (18)$$

Общая система уравнений принимает вид

$$\left. \begin{aligned} DI &= w_1 \cdot DH, \\ D\omega &= w_2 \cdot DF, \end{aligned} \right\} \quad (19)$$

а их решения определяются через псевдообратные матрицы

$$\left. \begin{aligned} w_1 &= DI \cdot DH^+, \\ w_2 &= D\omega \cdot DF^+. \end{aligned} \right\} \quad (20)$$

Следующий метод (третий случай) разработан для идентификации весовых коэффициентов второго нейрона для режима, когда часть элементов вектора состояния равна нулю, для этого при вращении двигателя с заданной скоростью происходит разрыв якорной цепи двигателя (ток якоря равен нулю) и осуществляется свободный выбег двигателя. Тогда второе уравнение системы (1) принимает вид

$$J \frac{d\omega}{dt} = -M_c(\omega). \quad (21)$$

Рассчитаем весовые коэффициенты второго нейрона в два этапа. На первом этапе разгоняется двигатель до номинальных значений и разрывается якор-

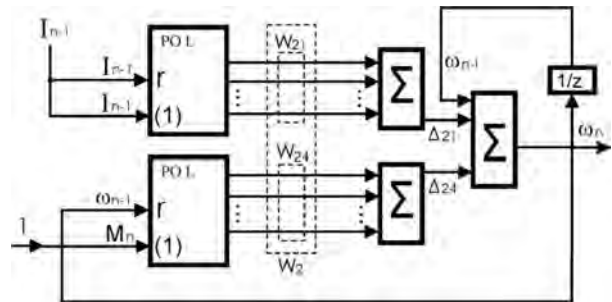


Рис. 2. Структура 2-го нейрона модели ДПТПВ в виде ПРНС

ная цепь, тогда сигналы части ПРНС, где формируется слагаемое Δ_{21} (рис. 2), равны нулю. В этом случае выход второго нейрона зависит только от одного нелинейного коэффициента (формируемого вектором весовых коэффициентов w_{24}), который вычисляется по данным режима.

На втором этапе используются данные режима разгона двигателя, по которым находим w_{21} по формуле

$$\Delta_{21} = \Delta\omega_n - \Delta_{24}. \quad (22)$$

Далее по второй формуле системы (20) рассчитываются весовые коэффициенты w_{21} . Аналогично для идентификации внутренних параметров первого уравнения системы (1) (весовых коэффициентов первого нейрона) необходимо задание двух следующих режимов: в первом режиме напряжение U и угловая скорость ω равны нулю (например, при подаче и отключении напряжения и фиксации ротора двигателя), во втором – только напряжение U равно нулю.

РЕЗУЛЬТАТЫ МОДЕЛИРОВАНИЯ

Расчет модели в виде ПРНС и идентификация нелинейных параметров выполнялись для ДПТПВ типа МП-62, имеющего следующие параметры [8]: $U_n = 220$ В; $I_n = 260$ А; $w_n = 53,4$ с⁻¹; $c = 78,5$; момент инерции, приведенный к валу двигателя, $J_D = 1,36$ кг·м². Моделирование выполнялось с шагом $T = 0,0001$ с. Переменными режима являлись: напряжение, подаваемое на двигатель, ток и угловая скорость двигателя.

Задавалось напряжение питания двигателя в виде, показанном на рис. 3. При такой форме напряжения существуют моменты времени, когда мгновенные значения угловой скорости одинаковы, а мгновенные значения токов двигателя разные. Происходит многократное прохождение одних и тех же мгновенных значений части координат объекта, что повышает точность идентификации.

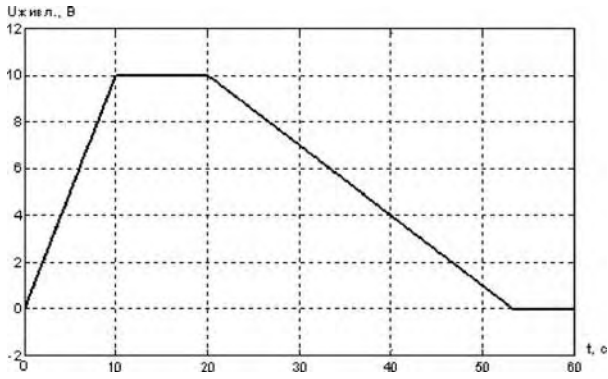


Рис. 3. Напряжение питания двигателя

По данным режима работы двигателя по уравнениям системы (8) рассчитывались весовые коэффициенты его модели в виде ПРНС. Графики токов и угловых скоростей для объекта I_{object} , ω_{object} и его модели I_{model} , ω_{model} приведены на рис. 4. Достигалась высокая точность математической модели (максимальная погрешность не превышает 1%). В математической модели с высокой точностью отработывался каждый из нелинейных членов дифференциального уравнения.

Изменения во времени идентифицированных с помощью ПРНС $c\Phi(I)$ и $M_c(\omega)$, согласно рис. 5, практически совпадают с изменениями этих величин в объекте. Максимальная ошибка также не превышает 1%.

Для исследования второго случая нелинейные зависимости момента сопротивления от скорости $M_c(\omega)$ задавались табл. 1. Нелинейности $\Phi(I)$ и $L(I)$ задавались в виде табл. 2 [8].

Графики тока и скорости для объекта и его модели в виде ПРНС приведены на рис. 6. Максимальная погрешность не превышала 5%. Для сравнения на рис. 7 приведены нелинейные графики изменения $c\Phi(I)$ и $M_c(\omega)$ в объекте и его модели в виде ПРНС. Погрешность идентификации при угловой скорости более 10 c^{-1} не превышала 25%.

Из рис. 6 и рис. 7 видно, что модель аппроксимирует нелинейности, но с большой погрешностью. Для повышения точности идентификации нужно повысить степень полинома. Если не использовать дополнительную информацию путем дифференцирования

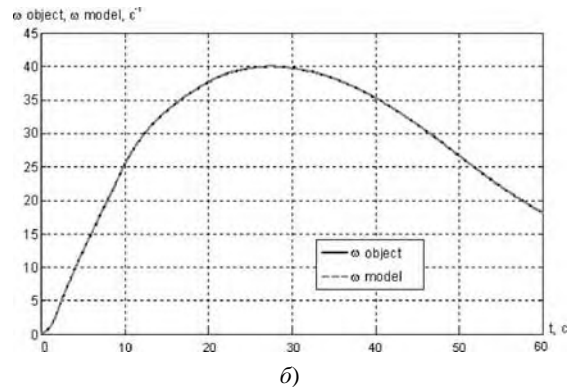
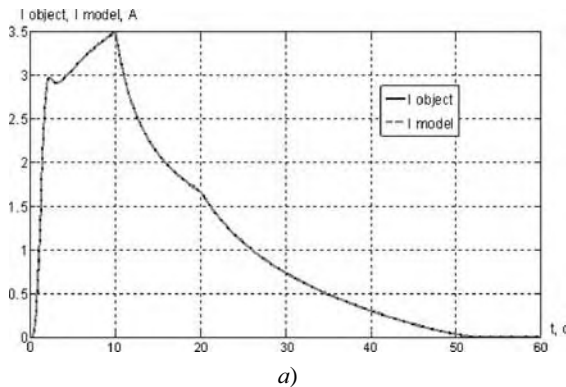


Рис. 4. Токи и угловые скорости двигателя в объекте и в ПРНС (первый случай)

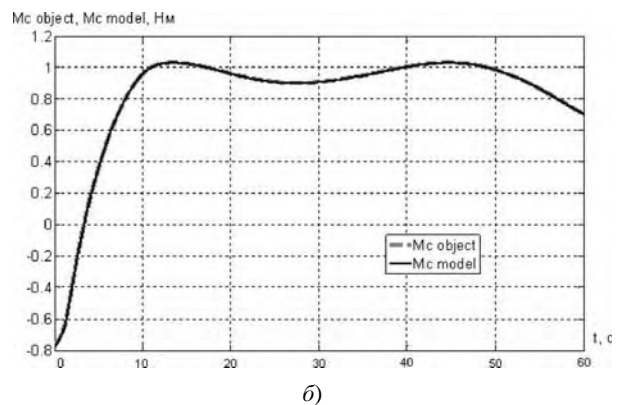
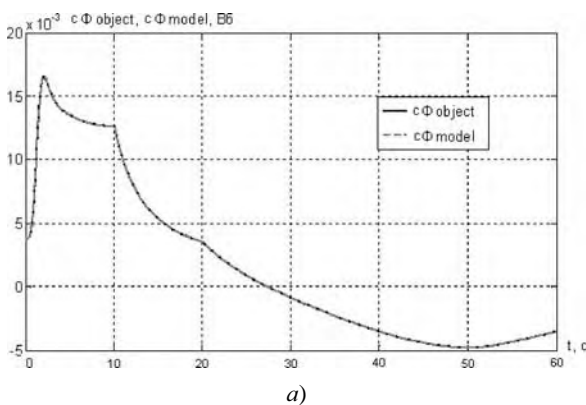


Рис. 5. Изменения $c\Phi(I)$ и $M_c(\omega)$ в объекте и в ПРНС (первый случай)

Таблиця 1. Зависимость момента сопротивления от скорости двигателя

Параметры, единицы измерения	Значения									
$\omega, \text{с}^{-1}$	0	0,1	20	40	60	80	100	120	140	150
$M_c, \text{Нм}$	0	0,7	0,75	0,82	0,9	1	1,1	1,2	1,3	1,35

Таблиця 2. Зависимости $\Phi(I)$ и $L(I)$

$I, \text{А}$	20	40	60	80	100	120	140	160	180	200	220	240	260	280	300	320
$\Phi, \text{Вб} \cdot 10^{-3}$	7	13,5	20,4	26,4	31,6	35,1	37,9	41	42,6	44,1	45,5	46,8	48	49,1	50,1	51
$L, \text{Гн} \cdot 10^{-3}$	36	34,5	32	28,5	24	19,5	15	11	9	7,5	6,3	5,5	5	4,5	4,2	4

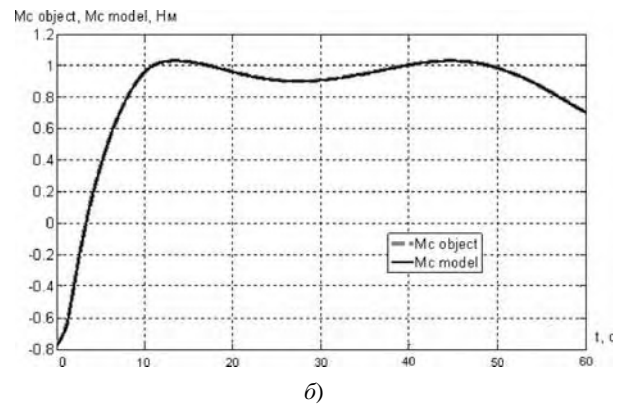
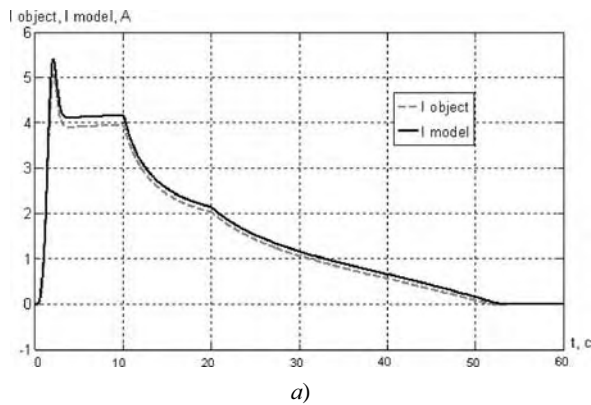


Рис. 6. Ток и угловая скорость двигателя в объекте и в ПРНС (второй случай)

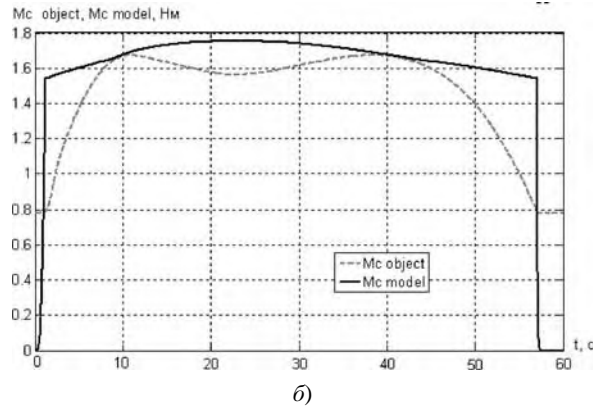
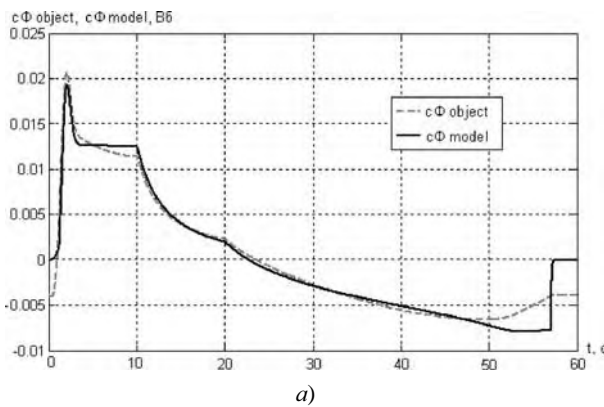


Рис. 7. Изменения $c\Phi(I)$ и $M_c(\omega)$ в объекте и в ПРНС (второй случай)

ния и осуществлять только запуск двигателя, ошибка идентификации нелинейностей значительно возрастает (до 50 % и выше).

Моделирование третьего случая выполнено для идентификации параметров второго уравнения системы (1) при нелинейных $\Phi(I)$ и $M_c(\omega)$, заданных в виде табл. 1 и 2. Для осуществления движения при токе двигателя, равном нулю, после разгона разрывалась цепь питания двигателя. Весовые коэффици-

циенты w_{24} рассчитывались по данным режима при свободном выбеге двигателя. Весовые коэффициенты w_{21} рассчитывались по данным режима при разгоне двигателя с использованием уравнения (22).

Идентифицированные из ПРНС нелинейности $c\Phi(I)$ и $M_c(\omega)$ (рис. 8) очень близки к нелинейностям, заданным в объекте. Максимальная погрешность идентификации не превышает 5 %.

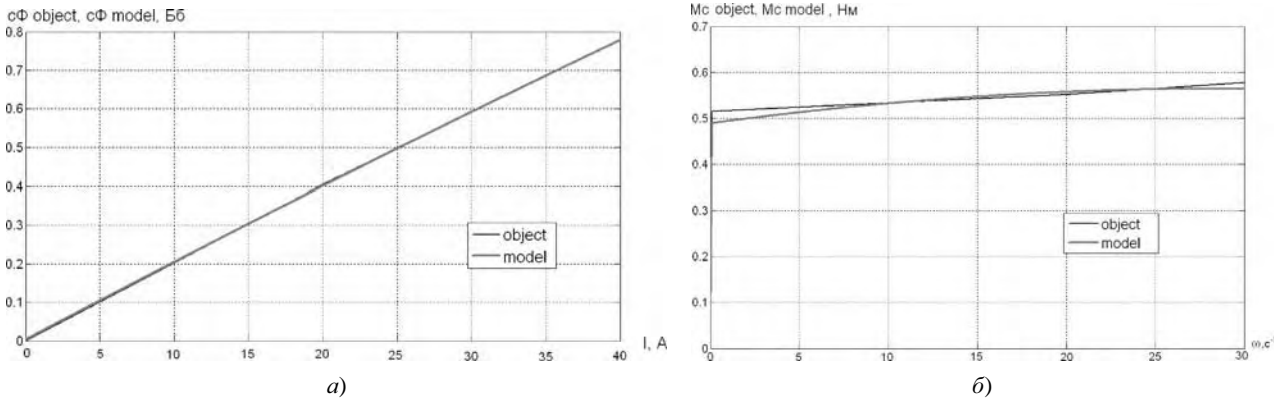


Рис. 8. Графики $c\Phi(I)$ и $M_c(\omega)$ в объекте и их идентификация из ПРНС (третий случай)

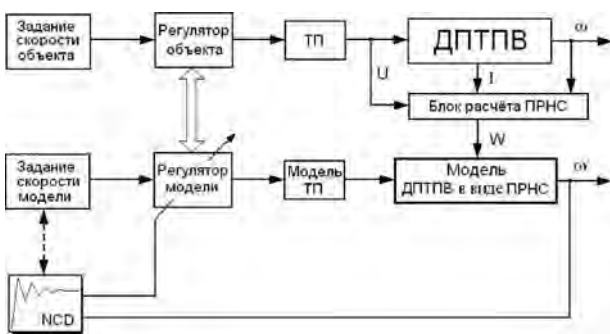


Рис. 9. Структурная схема настройки СУ ЭП с ДПТПВ

Таблица 3. Ошибки обработки тока и угловой скорости

Наименование параметра	Максимальные ошибки, %	Ошибки в установившемся режиме, %
Ток якоря (I)	3	0,72
Скорость (ω)	2	0,28

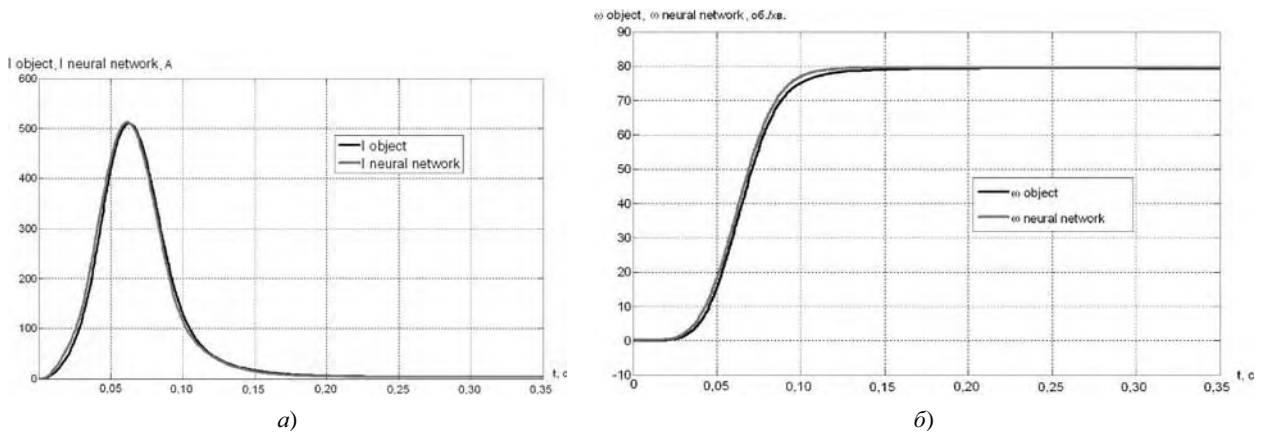


Рис. 10. Графики переходных процессов тока и угловой скорости после настройки регулятора скорости

НАСТРОЙКА СУ ЭП ПО ЕГО МАТЕМАТИЧЕСКОЙ МОДЕЛИ В ВИДЕ ПРНС

На синтезированной выше математической модели ДПТПВ в виде ПРНС, полученной для более общего второго случая, выполнялась настройка регулятора скорости ЭП. Последовательность настройки следующая (рис. 9): синтез математической модели ДПТПВ в виде ПРНС; настройка интегрального k_i и

дифференциального k_n параметров ПИ-регулятора на этой математической модели; подстановка найденных параметров k_i и k_n в регулятор объекта; проверка качества переходных процессов при обработке задания ЭП с найденными коэффициентами регулятора.

В блоке «Блок расчета ПРНС» (рис. 9) по данным режима работы двигателя рассчитываются весовые коэффициенты модели ДПТПВ в виде ПРНС. В блоке «NCD» [9] задаются границы желаемого переход-

ного процесса и настраиваются параметры регулятора на математической модели объекта в виде ПРНС. При изменении параметров объекта синтезируется новая модель в виде ПРНС и на ней перенастраивается регулятор, что обеспечивает работу ЭП с желаемыми характеристиками переходного процесса.

Ошибка обработки координат ЭП моделью в виде ПРНС на обучающих наборах не превышала 3 % (рис. 10). Переходные процессы тока и угловой скорости исследуемого объекта (I_{object} , ω_{object}) и его модели в виде ПРНС ($I_{neural\ network}$, $\omega_{neural\ network}$) с найденными параметрами регуляторов не имеют перерегулирования и при этом практически одинаковы, из этого следует, что найденная модель ДПТПВ в виде ПРНС имеет необходимую точность и обобщающие свойства для настройки регуляторов.

В табл. 3 приведены максимальные значения ошибок и значения ошибок в установившемся режиме обработки тока и угловой скорости ЭП с моделью ДПТПВ в виде ПРНС.

Автор благодарен магистру Смирнову А. С. за совместную работу при выполнении моделирования.

ВЫВОДЫ

1. По результатам исследований выполнено сравнение предложенных способов идентификации параметров ДПТПВ по его модели в виде ПРНС. Способ первый позволяет получить высокую точность идентификации (погрешность 1 %), но применяется при нелинейностях, точно аппроксимируемых полиномами заданной степени. Способ второй применяется для нелинейности, заданной произвольно (например, таблицей), однако требует усложнения вычислений и обеспечивает невысокую точность (погрешность 25 %). В третьем способе обеспечивается высокая точность (5 %), но требуется создание специальных режимов двигателя (разрыв цепи якоря).

2. Выполненная настройка параметров ПИ-регулятора скорости по полученной математической модели ДПТПВ в виде ПРНС позволила получить в ЭП желаемые переходные процессы, которые практически совпадают для модели и объекта, следовательно, синтезированная модель ДПТПВ в виде ПРНС имеет необходимую точность и обобщающие свойства для настройки регуляторов.

СПИСОК ЛИТЕРАТУРЫ

1. Бодянский, Е. В. Искусственные нейронные сети: архитектура, обучение, применения / Е. В. Бодянский, О. Г. Руденко // Харьков : ТЕЛЕТЕХ, 2004. – 372 с.
2. Хайкин, С. Нейронные сети: полный курс / Хайкин С. – 2-е издание. – М. : Вильямс, 2006. – 1104 с.

3. Омату, С. Нейроуправление и его приложения. Кн. 2. / С. Омату, М. Халид, Р. Юсоф ; пер. с англ. Н. В. Батина. – М. : ИПРЖР, 2000. – 272 с. – (Нейрокомпьютеры и их применение).
4. Генетические алгоритмы, искусственные нейронные сети и проблемы виртуальной реальности / Г. К. Вороновский, К. В. Махотило, С. Н. Петрашов, С. А. Сергеев. – Харьков : ОСНОВА, 1997. – 112 с.
5. Орловский, И. А. Расчет моделей нелинейных электро-механических объектов на полиномиальных рекуррентных нейронных сетях из их известных математических моделей / И. А. Орловский // Технічна електродинаміка. – 2009. – № 1. – С. 20–31.
6. Орловский, И. А. Разработка моделей на нейронных сетях серийного двигателя постоянного тока по экспериментальным данным / И. А. Орловский, И. В. Блохин, А. С. Смирнов // Вісник Кременчуцького політехнічного університету. – Кременчук. – 2010. – Вип. 3 (62), част. 1. – С. 153–156.
7. Орловский, И. А. Обучение на рекуррентной нейронной сети модели серийного двигателя и идентификация его внутренних параметров. / И. А. Орловский, И. В. Блохин // Вісник Національного технічного університету «Харківський політехнічний інститут» збірник наукових трудов «Проблеми автоматизованого приводу. Теорія і практика». – 2010. – № 28. – С. 566–568.
8. Андреев, В. П. Основы электропривода / В. П. Андреев, Ю. А. Сабинин. – М.-Л. : Госэнергоиздат, 1963. – 772 с.
9. Дьяконов, В. П. МАТЛАБ 5.3.1 с пакетами расширений / под ред. проф. В. П. Дьяконова ; В. П. Дьяконов, И. В. Абраменкова, В. В. Круглов. – М. : Нолидж, 2001. – 880 с.

Надійшла 15.07.2010

Орловський І. А.

ІДЕНТИФІКАЦІЯ МАТЕМАТИЧНОЇ МОДЕЛІ У ВИГЛЯДІ ПОЛІНОМІАЛЬНОЇ РЕКУРЕНТНОЇ НЕЙРОННОЇ МЕРЕЖІ ТА НАСТРОЙКА РЕГУЛЯТОРІВ ЕЛЕКТРОПРИВОДУ З СЕРІЕСНИМ ДВИГУНОМ

Синтезовано математичні моделі електропривода з серіесним двигуном у вигляді поліноміальних рекуррентних нейронних мереж (ПРНМ) за даними режиму його роботи. Досліджено способи ідентифікації параметрів приводу і залежності моменту опору від швидкості двигуна при заданні спеціальних режимів роботи і різних видів рівнянь, що описують нелінійності параметрів. Виконано настройку ПІ-регулятора швидкості на отриманій моделі у вигляді ПРНМ.

Ключові слова: математична модель, двигун постійного струму послідовного збудження, рекуррентна нейронна мережа, регулятор швидкості, ідентифікація параметрів.

Orlovsky I. A.

IDENTIFICATION OF MATHEMATICAL MODEL IN THE FORM OF POLYNOMIAL RECURRENT NEURAL NETWORK AND ADJUSTMENT OF ELECTRIC DRIVE WITH SERIES-WOUND MOTOR

Mathematical models of electric drive with a series-wound motor in the form of polynomial recurrent neural networks (PRNN) have been synthesized using its operational data. The methods of identification of drive parameters and drag torque dependence on motor speed were studied, setting special operating modes and different types of equations describing parameters non-linearity. A PI speed controller was adjusted according to the obtained model in the form of PRNN.

Key words: mathematical model, DC motor of series excitation, recurrent neural network, speed controller, parameters identification.

¹Канд. техн. наук, доцент Национального университета кораблестроения им. адмирала Макарова, г. Николаев

²Д-р техн. наук, профессор Национального университета кораблестроения им. адмирала Макарова, г. Николаев

СИНТЕЗ СТРУКТУРНО ПЕРЕКЛЮЧАЕМЫХ СИСТЕМ ДЛЯ УПРАВЛЕНИЯ МНОГОМЕРНЫМИ ПОДВИЖНЫМИ ОБЪЕКТАМИ

В статье рассматривается применение метода структурно переключаемых обратных связей для синтеза управляющих функций в замкнутых системах управления многомерными подвижными объектами. Сформированы принципы построения оптимальной траектории движения объекта и определения моментов переключения управляющих функций в цепях обратных связей, получены значения вектора управления для замкнутых систем различного типа.

Ключевые слова: многомерные подвижные объекты, структурно переключаемые обратные связи, замкнутые системы управления.

ВВЕДЕНИЕ

Одним из базовых принципов синтеза систем управления подвижными объектами являются подходы, основанные на решении обратных задач динамики (ОЗД) объектов управления, т. к. динамика подвижных объектов, например, морских подвижных объектов, достаточно исследована и, таким образом, на базе математической модели установлена структурно-параметрическая связь между фазовыми координатами объекта, что позволяет максимально использовать это знание при синтезе управления.

ОЗД формировались и решались еще У. Гамильтоном, Ж. Лагранжем в задачах аналитической механики, позже в работах по механике управляемого полета и других приложениях Н. Еругина, А. Галиуллина, а также в работах последнего времени при синтезе систем с обратными связями [1, 2]. Систематизировано и детально рассмотрено решение ОЗД управляемых систем в работах [3, 4] для линейных и нелинейных моделей, в т. ч. при синтезе робастно устойчивых структур управляемых систем высокой динамической точности [5].

При всем многообразии требований, которые предъявляются к системам управления, можно при одинаковых требованиях к точности управления выделить два основных физических качества – минимум расхода энергии при управлении и максимальное быстродействие для переходного процесса управления [6]. Эти требования, естественно, являются экстремальными, а система, которая отвечает им, классически оптимальна в том смысле, как это определяется с момента возникновения теории оптимального управления.

ПОСТАНОВКА ЗАДАЧИ ИССЛЕДОВАНИЯ

Для класса подвижных объектов наибольший интерес представляет создание инженерных методов исследования показателей качества многомерных динамических моделей и синтеза оптимальных управляющих воздействий. Аналитическое решение краевых задач при использовании классических методов оптимального управления с учетом многомерности объекта управления сложно и громоздко, а численное решение в ряде случаев не дает достаточно быстрой сходимости. В этой связи существует потребность в развитии инженерных методов оптимального управления, которые не нуждаются в непосредственном решении дифференциальных уравнений динамики объекта.

Метод структурно переключаемых обратных связей [7–9] позволяет решать практические задачи построения оптимальной траектории движения для разных критериев оптимальности и вида граничных условий. При этом движение по отрезкам оптимальной траектории обеспечивается управляющими функциями в цепях обратной связи, которые имеют возможность переключения для перехода с текущего отрезка траектории движения на заданный отрезок.

Данный метод состоит из следующих основных этапов [7]:

- планирование оптимальной траектории;
- определение моментов переключения управляющих функций в цепях обратной связи объекта;
- синтез управляющих функций в соответствующих цепях обратной связи многомерного объекта.

Планирование траектории перехода объекта для заданных критериев (условий) оптимальности и граничных условий заключается в определении необхо-

димого количества отрезков траектории с постоянными значениями соответствующих производных фазовых переменных, а также моментов времени переключения управляющих функций в цепях обратных связей при переходе с начального отрезка на заданный отрезок траектории. Переключение управляющих функций изменяет структурную конфигурацию обратных связей и решает задачу обеспечения необходимого порядка производной фазовой переменной с соответствующим позитивным или негативным постоянным значением. Синтез управляющих функций при этом будет заключаться в определении управляющих воздействий в цепях обратных связей объекта управления, при которых на определенных отрезках фазовой траектории выполняются условия постоянства соответствующих производных фазовых переменных. Произвольную фазовую траекторию объекта управления можно описать как совокупность отрезков траектории с постоянной производной определенного порядка. Планирование фазовой траектории объекта управления для заданных начальных и конечных условий включает определение необходимого количества отрезков фазовой траектории с постоянными значениями соответствующих производных вектора фазовых координат, а также вычисление моментов переключения управлений в обратных связях.

Целью данной статьи является разработка методов решения задач, стоящих перед перечисленными выше этапами синтеза замкнутых систем управления многомерными подвижными объектами.

Для i -го отрезка фазовой траектории произвольного (в т. ч. нелинейного, нестационарного) динамического объекта запишем, применяя разложение в ряд Тейлора,

$$\begin{aligned} \mathbf{X}(t_{i+1}) = & \mathbf{X}(t_i) + \frac{d\mathbf{X}(t)}{dt} \Big|_{t_i} \frac{\Delta t^1}{1!} + \dots + \\ & + \frac{d^k \mathbf{X}(t)}{dt^k} \Big|_{t_i} \frac{(\Delta t^l)^k}{k!} + \dots + \mathbf{R}_p, \end{aligned} \quad (1)$$

где $\mathbf{X}(t) = \{x_x(t); x_y(t); x_z(t)\}$ – вектор фазовых координат движения по координатным осям $l = \{x, y, z\}$; \mathbf{R}_p – вектор, определяющий остаточные члены ряда Тейлора; $\Delta t^l = t_{i+1}^l - t_i^l$; t_i^l, t_{i+1}^l – начальный и конечный моменты времени движения объекта по каждой из координатных осей на i -м отрезке траектории.

Планирование фазовой траектории объекта управления для, соответственно, начальных и конечных условий

$$\begin{aligned} & \left\{ x_{ox}^{(k)}, x_{oy}^{(k)}, x_{oz}^{(k)}; \dot{x}_{ox}^{(k)}, \dot{x}_{oy}^{(k)}, \dot{x}_{oz}^{(k)}; \dots; x_{ox}^{(k)}, x_{oy}^{(k)}, x_{oz}^{(k)} \right\}; \\ & \left\{ x_{fx}^{(m)}, x_{fy}^{(m)}, x_{fz}^{(m)}; \dot{x}_{fx}^{(m)}, \dot{x}_{fy}^{(m)}, \dot{x}_{fz}^{(m)}; \dots; x_{fx}^{(m)}, x_{fy}^{(m)}, x_{fz}^{(m)} \right\} \end{aligned}$$

заключается в определении необходимого количества отрезков фазовой траектории с постоянными значениями соответствующих производных вектора $\mathbf{X}(t)$, а также моментов времени t_i^l переключения управляющих функций в обратных связях для обеспечения перехода объекта с данного отрезка траектории на последующий отрезок по каждой из координатных осей $l = \{x, y, z\}$.

Синтез оптимальных управляющих функций для многомерного объекта с применением метода структурно переключаемых обратных связей рассмотрим на примере задачи управления с минимумом расхода энергии. Рассмотрим уравнения движения динамического объекта на горизонтальной плоскости при равных граничных условиях по двум возможным фазовым траекториям:

$$\mathbf{X}(T_1) = \mathbf{X}(0) + \dot{\mathbf{X}}(0)T_1 + \ddot{\mathbf{X}}(0) \frac{T_1^2}{2!} + \dots + \mathbf{X}^{(m)}(0) \frac{T_1^m}{m!}, \quad (2)$$

$$\mathbf{X}(T_2) = \mathbf{X}(0) + \dot{\mathbf{X}}(0)T_2 + \ddot{\mathbf{X}}(0) \frac{T_2^2}{2!} + \dots + \mathbf{X}^{(r)}(0) \frac{T_2^r}{r!}, \quad (3)$$

где $\mathbf{X}(0), \mathbf{X}(T_1) = \mathbf{X}(T_2)$ – векторы начальных и конечных координат; T_1, T_2 – время движения по каждой из траекторий; m, r – порядок высших производных вектора фазовых координат.

Оценка кинетической энергии, затраченной на движение, показывает, что для траектории, определяемой уравнением (3), при выполнении условия $m > r$ расход энергии меньше по сравнению с траекторией (2). При этом в случае $(m - r) = 1$ разница кинетических энергий $\Delta E = \min$, а с ростом $(m - r)$ величина ΔE будет также возрастать. Последующий анализ показывает, что оптимальной по минимуму расхода энергии является траектория с наименьшим количеством минимальных по амплитуде производных фазовых переменных, необходимых для выполнения всех граничных условий.

ОПРЕДЕЛЕНИЕ МОМЕНТОВ ПЕРЕКЛЮЧЕНИЯ УПРАВЛЯЮЩИХ ФУНКЦИЙ В ЦЕПЯХ ОБРАТНЫХ СВЯЗЕЙ ОБЪЕКТА УПРАВЛЕНИЯ

Переключение управляющих функций обеспечивает требуемый порядок производной фазовой координаты с соответствующим положительным или отрицательным постоянным значением. Синтез управля-

ющих функций будет состоять в определении управляющих воздействий в цепях обратных связей объекта управления, при которых на определенных заданных отрезках фазовой траектории выполняются условия постоянства соответствующих производных фазовой переменной $\mathbf{X}(t)$. Таким образом, при движении по координатным осям $l = \{x, y, z\}$ на каждом отрезке фазовой траектории будет включаться на определенное время Δt^l соответствующая цепь обратной связи, реализующая заданную управляющую функцию.

Моменты переключения управляющих функций в обратной связи будут определяться на основе энергетического анализа (гамильтониана системы), вида планируемых фазовых траекторий с учетом ограничений на управляющее воздействие, необходимых условий оптимальности и заданных граничных условий.

В общем виде процесс управления можно разбить на этапы «разгона» и «торможения» до заданных параметров, например, постоянной скорости движения.

Фазовые уравнения процесса «разгона» до заданной скорости $\dot{\mathbf{X}}(t_{i+1})$ с последующим движением объекта управления с ее постоянным значением $\dot{\mathbf{X}}(t_{i+1}) = \text{const}$ до момента времени $(t_i + 2)$ запишем в виде

$$\begin{aligned} \mathbf{X}(t_1) &= \mathbf{X}(0) + \dot{\mathbf{X}}(0)t_1 + \ddot{\mathbf{X}}(0)\frac{t_1^2}{2} + \dots + \mathbf{X}^{(k)}(0)\frac{t_1^k}{k!}; \\ &\dots\dots\dots \\ \mathbf{X}(t_{i+1}) &= \mathbf{X}(t_i) + \dot{\mathbf{X}}(t_i)(t_{i+1} - t_i) + \ddot{\mathbf{X}}(t_i)\frac{(t_{i+1} - t_i)^2}{2}; \\ \mathbf{X}(t_{i+2}) &= \mathbf{X}(t_{i+1}) + \dot{\mathbf{X}}(t_{i+1})(t_{i+2} - t_{i+1}). \end{aligned}$$

Уравнения этапа «торможения» от постоянной скорости $\dot{\mathbf{X}}(t_{i+1}) = \text{const}$ до нового (заданного) значения скорости $\dot{\mathbf{X}}(t_m) = \text{const}$ будут иметь вид

$$\begin{aligned} \mathbf{X}(t_{i+3}) &= \mathbf{X}(t_{i+2}) + \dot{\mathbf{X}}(t_{i+2})(t_{i+3} - t_{i+2}) - \dots - \\ &\quad - \mathbf{X}^{(k)}(t_{i+2})\frac{(t_{i+3} - t_{i+2})^k}{k!}; \\ &\dots\dots\dots \\ \mathbf{X}(t_m) &= \mathbf{X}(t_{m-1}) + \dot{\mathbf{X}}(t_{m-1})(t_m - t_{m-1}) - \\ &\quad - \ddot{\mathbf{X}}(t_{m-1})\frac{(t_m - t_{m-1})^2}{2}; \\ \mathbf{X}(t_{m+1}) &= \mathbf{X}(t_m) + \dot{\mathbf{X}}(t_m)(t_{m+1} - t_m). \end{aligned}$$

При этом t_i – моменты переключения управляющих функций в цепях обратных связей ($i = 1, \dots, m$).

СИНТЕЗ СТРУКТУРНО ПЕРЕКЛЮЧАЕМЫХ СИСТЕМ УПРАВЛЕНИЯ

Общий порядок синтеза структурно переключаемых систем на основе обратных связей рассмотрен в работах [7, 8]. В данной статье предложенный подход расширяется для замкнутых систем различного вида. При этом обратная связь, замыкающая систему, будет являться главной, а формируемые на основе предложенного метода структурно переключаемые обратные связи – дополнительными. Для управляемой системы (рис. 1), замкнутой единичной обратной связью, динамика описывается векторно-матричным уравнением вида

$$\dot{\mathbf{X}}(t) = \mathbf{A}\mathbf{X}(t) + \mathbf{B}[\mathbf{U}(t) - \mathbf{X}(t)] = (\mathbf{A} - \mathbf{B})\mathbf{X}(t) + \mathbf{B}\mathbf{U}(t). \quad (4)$$

После дифференцирования уравнения (4) получим вторую производную

$$\ddot{\mathbf{X}}(t) = (\mathbf{A} - \mathbf{B})\dot{\mathbf{X}}(t) + \mathbf{B}\dot{\mathbf{U}}(t), \quad (5)$$

а после повторения процедуры дифференцирования – третью производную

$$\ddot{\ddot{\mathbf{X}}}(t) = (\mathbf{A} - \mathbf{B})\ddot{\mathbf{X}}(t) + \mathbf{B}\ddot{\mathbf{U}}(t). \quad (6)$$

Для траектории вида (2) при $\ddot{\mathbf{X}}(t) = 0$ после подстановки последовательно уравнения (4) в (5) и дальше в (6) получим уравнение

$$\begin{aligned} \mathbf{B}\ddot{\mathbf{U}}(t) &= -(\mathbf{A} - \mathbf{B})^3\mathbf{X}(t) - \\ &\quad - (\mathbf{A} - \mathbf{B})^2\mathbf{B}\mathbf{U}(t) - (\mathbf{A} - \mathbf{B})\mathbf{B}\dot{\mathbf{U}}(t). \end{aligned} \quad (7)$$

После использования преобразования Лапласа с учетом начального значения вектора управляющих функций \mathbf{U}_0 и векторно-матричных преобразований получаем

$$\begin{aligned} \mathbf{U}(p) &= -\mathbf{B}^{-1}[p^2\mathbf{E} + p(\mathbf{A} - \mathbf{B}) + (\mathbf{A} - \mathbf{B})^2]^{-1} \times \\ &\quad \times \{(\mathbf{A} - \mathbf{B})^3\mathbf{X}(p) - (p\mathbf{B} + [(\mathbf{A} - \mathbf{B})\mathbf{B}]\mathbf{U}_0 + \mathbf{B}\dot{\mathbf{U}}_0)\}. \end{aligned}$$

Для случая, когда обратная связь не является единичной, а описывается матрицей постоянных пара-

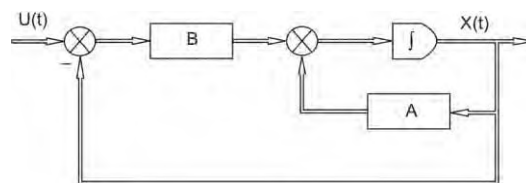


Рис. 1. Структурная схема многомерной замкнутой системы

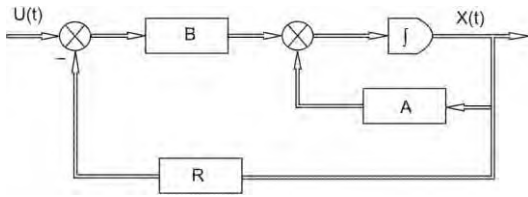


Рис. 2. Структурная схема замкнутой многомерной системы с обратной связью, описываемой матрицей \mathbf{R}

метров \mathbf{R} (рис. 2), векторно-матричное уравнение (4) запишется в виде

$$\dot{\mathbf{X}}(t) = (\mathbf{A} - \mathbf{BR})\mathbf{X}(t) + \mathbf{BU}(t). \quad (8)$$

Дифференцирование уравнения (8) дает уравнение управляемой системы для траектории движения объекта вида (2) с учетом выполнения условия $\dot{\mathbf{X}}(t) = 0$

$$\mathbf{B}\dot{\mathbf{U}}(t) = -(\mathbf{A} - \mathbf{BR})^2 \mathbf{X}(t) - (\mathbf{A} - \mathbf{BR})\mathbf{BU}(t).$$

При использовании преобразования Лапласа с учетом начального значения вектора управляющих функций \mathbf{U}_0 запишем для вектора управлений:

$$\mathbf{U}(p) = -\mathbf{B}^{-1}(p\mathbf{E} + \mathbf{A} - \mathbf{BR})^{-1} \times [(\mathbf{A} - \mathbf{BR})^2 \mathbf{X}(p) - \mathbf{BU}_0].$$

После повторения процедуры дифференцирования (8) получим уравнение динамики при $\dot{\mathbf{X}}(t) = 0$:

$$\mathbf{B}\ddot{\mathbf{U}}(t) = -(\mathbf{A} - \mathbf{BR})^3 \mathbf{X}(t) - (\mathbf{A} - \mathbf{BR})^2 \mathbf{BU}(t) - (\mathbf{A} - \mathbf{BR})\mathbf{B}\dot{\mathbf{U}}(t);$$

а также, с использованием необходимых преобразований, выражение для вектора управлений:

$$\mathbf{U}(p) = -\mathbf{B}^{-1}[p^2\mathbf{E} + p(\mathbf{A} - \mathbf{BR}) + (\mathbf{A} - \mathbf{BR})^2]^{-1} \times \{(\mathbf{A} - \mathbf{BR})^3 \mathbf{X}(p) - [p\mathbf{B} + (\mathbf{A} - \mathbf{BR})\mathbf{B}]\mathbf{U}_0 + \mathbf{B}\dot{\mathbf{U}}_0\}.$$

В случае переменной обратной связи динамическая система (8) становится нестационарной вида

$$\begin{aligned} \dot{\mathbf{X}}(t) &= \mathbf{A}\mathbf{X}(t) + \mathbf{B}[\mathbf{U}(t) - \mathbf{R}(t)\mathbf{X}(t)] = \\ &= [\mathbf{A} - \mathbf{BR}(t)]\mathbf{X}(t) + \mathbf{BU}(t), \end{aligned} \quad (9)$$

т. к. матрица $\mathbf{R}(t)$ будет включать компоненты, которые являются функциями времени. Проведем дифференцирование уравнения (9) и получим, например, для второй производной

$$\begin{aligned} \ddot{\mathbf{X}}(t) &= [-\dot{\mathbf{B}}\mathbf{R} + (\mathbf{A} - \mathbf{BR})^2]\mathbf{X}(t) + \\ &+ (\mathbf{A} - \mathbf{BR})\mathbf{BU}(t) + \mathbf{B}\dot{\mathbf{U}}(t). \end{aligned} \quad (10)$$

Далее для третьей производной запишем

$$\begin{aligned} \dddot{\mathbf{X}}(t) &= [-\ddot{\mathbf{B}}\mathbf{R} - 2\dot{\mathbf{B}}\dot{\mathbf{R}} - (\mathbf{A} - \mathbf{BR})\dot{\mathbf{B}}\mathbf{R} + \\ &+ (\mathbf{A} - \mathbf{BR})^3]\mathbf{X}(t) + [-2\dot{\mathbf{B}}\dot{\mathbf{R}} + (\mathbf{A} - \mathbf{BR})^2\mathbf{B}]\mathbf{U}(t) + \\ &+ (\mathbf{A} - \mathbf{BR})\mathbf{B}\dot{\mathbf{U}}(t) + \mathbf{B}\ddot{\mathbf{U}}(t). \end{aligned} \quad (11)$$



Рис. 3. Структурная схема системы управления вида (8) с дополнительными обратными связями (БКП – блок ключей переключения)

При выполнении условия $\dot{\mathbf{X}} = 0$ будем иметь из (11)

$$[-2\dot{\mathbf{B}}\mathbf{R} + (\mathbf{A} - \mathbf{B}\mathbf{R})^2]\mathbf{B}\mathbf{U}(t) + (\mathbf{A} - \mathbf{B}\mathbf{R})\mathbf{B}\dot{\mathbf{U}}(t) + \mathbf{B}\ddot{\mathbf{U}}(t) = [\mathbf{B}\ddot{\mathbf{R}} + 2\dot{\mathbf{B}}\mathbf{R}(\mathbf{A} - \mathbf{B}\mathbf{R}) + (\mathbf{A} - \mathbf{B}\mathbf{R})\dot{\mathbf{B}}\mathbf{R} - (\mathbf{A} - \mathbf{B}\mathbf{R})^3]\mathbf{X}(t). \quad (12)$$

Для квазистационарных объектов с медленно изменяющимися параметрами примем, что $\mathbf{R}(t) = \mathbf{R}_{\Delta t_c}$, $\dot{\mathbf{R}}(t) = \dot{\mathbf{R}}_{\Delta t_c}$, $\ddot{\mathbf{R}}(t) = \ddot{\mathbf{R}}_{\Delta t_c}$ есть матрицы, компоненты которых известны, постоянны («заморожены» [9] на интервале Δt_c) и независимы для определенного интервала времени Δt_c . Тогда, применяя преобразование Лапласа, получим из уравнения (10) для вектора управления $\mathbf{U}(p)$ при выполнении условия $\dot{\mathbf{X}}(t) = 0$ с учетом нулевых начальных значений

$$\mathbf{U}(p) = -[p\mathbf{B} + (\mathbf{A} - \mathbf{B}\mathbf{R}_{\Delta t_c})\mathbf{B}]^{-1} \times [-\mathbf{B}\dot{\mathbf{R}}_{\Delta t_c} + (\mathbf{A} - \mathbf{B}\mathbf{R}_{\Delta t_c})^2]\mathbf{X}(p).$$

Решение преобразованного уравнения (12) относительно $\mathbf{U}(p)$ определяет выражение вектора управляющих функций для выполнения условия $\dot{\mathbf{X}} = 0$

$$\mathbf{U}(p) = \{p^2\mathbf{B} + p(\mathbf{A} - \mathbf{B}\mathbf{R}_{\Delta t_c})\mathbf{B} - [2\mathbf{B}\dot{\mathbf{R}}_{\Delta t_c} - (\mathbf{A} - \mathbf{B}\mathbf{R}_{\Delta t_c})^2]\mathbf{B}\}^{-1} \times [\mathbf{B}\ddot{\mathbf{R}}_{\Delta t_c} + 2\dot{\mathbf{B}}\mathbf{R}_{\Delta t_c}(\mathbf{A} - \mathbf{B}\mathbf{R}_{\Delta t_c}) + (\mathbf{A} - \mathbf{B}\mathbf{R}_{\Delta t_c})\mathbf{B}\dot{\mathbf{R}}_{\Delta t_c} - (\mathbf{A} - \mathbf{B}\mathbf{R}_{\Delta t_c})^3]\mathbf{X}(p).$$

Для случая, когда на вход стационарной системы (рис. 4) подается вектор наблюдаемых координат \mathbf{Y} , будем иметь векторно-матричное уравнение системы

$$\dot{\mathbf{X}}(t) = (\mathbf{A} - \mathbf{B}\mathbf{C})\mathbf{X}(t) + (\mathbf{B} - \mathbf{B}\mathbf{D})\mathbf{U}(t).$$

В результате соответствующих векторно-матричных преобразований вектор управляющих функций

$\mathbf{U}(p)$ при условии $\dot{\mathbf{X}}(t) = 0$ запишется в виде

$$\mathbf{U}(p) = -(\mathbf{B} - \mathbf{B}\mathbf{D})^{-1}(p^{k-1}\mathbf{E} + \mathbf{A}_k)^{-1} \times [\mathbf{A} - \mathbf{B}\mathbf{C}]^k \mathbf{X}(p) - \mathbf{U}_0(p), \quad (13)$$

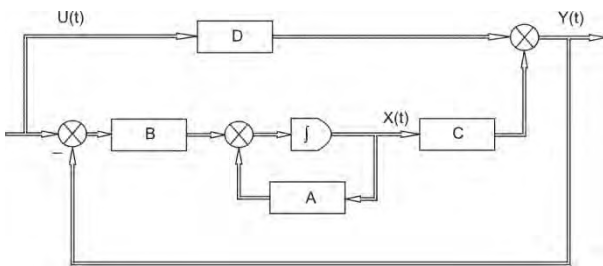


Рис. 4. Структурная схема многомерной системы управления с обратной связью по вектору выходных (наблюдаемых) координат

где матрица $\mathbf{A}_k = \sum_{j=0}^{k-2} (p^j \mathbf{A} - \mathbf{B}\mathbf{C})^{k-j-1}$; $\mathbf{U}_0(p)$ – начальный вектор управляющей функции.

При обратной связи, которая включает матрицу параметров \mathbf{R} , будем иметь уравнение (13) в виде

$$\mathbf{U}(p) = -(\mathbf{B} - \mathbf{B}\mathbf{R}\mathbf{D})^{-1}(p^{k-1}\mathbf{E} + \mathbf{A}_k)^{-1} \times [(\mathbf{A} - \mathbf{B}\mathbf{R}\mathbf{C})^k \mathbf{X}(p) - \mathbf{U}_0(p)].$$

При переменной обратной связи $\mathbf{R}(t)$ динамика управляемой системы примет вид

$$\dot{\mathbf{X}}(t) = [\mathbf{A} - \mathbf{B}\mathbf{R}(t)\mathbf{C}]\mathbf{X}(t) + [\mathbf{B} - \mathbf{B}\mathbf{R}(t)\mathbf{D}]\mathbf{U}(t). \quad (14)$$

Преобразование уравнения (14) для квазистационарной системы аналогично (10) и решение относительно $\mathbf{U}(p)$ определяют выражение вектора управляющих функций при условии $\dot{\mathbf{X}}(t) = 0$

$$\mathbf{U}(p) = \{p(\mathbf{B} - \mathbf{B}\mathbf{R}_{\Delta t_c} \mathbf{D}) + [(\mathbf{A} - \mathbf{B}\mathbf{R}_{\Delta t_c} \mathbf{C})(\mathbf{B} - \mathbf{B}\mathbf{R}_{\Delta t_c} \mathbf{D}) - \mathbf{B}\dot{\mathbf{R}}_{\Delta t_c} \mathbf{D}]\}^{-1} \times [\mathbf{B}\dot{\mathbf{R}}_{\Delta t_c} \mathbf{C} - (\mathbf{A} - \mathbf{B}\mathbf{R}_{\Delta t_c} \mathbf{C})^2]\mathbf{X}(p).$$

Полученные значения управляющих функций дают возможность для разных типов динамических систем обеспечить движение многомерного объекта по заданной оптимальной траектории движения.

ВЫВОДЫ

Предложенный метод структурно переключаемых обратных связей для синтеза управляющих воздействий позволяет для широкого класса многомерных подвижных объектов решать практические задачи построения оптимальных фазовых траекторий для различных условий оптимальности движения и вида граничных условий. Данный подход не требует непосредственного измерения производных фазовых переменных на выходе объекта управления, а определение моментов переключения управляющих функций осуществляется путем решения системы алгебраических уравнений. Полученные решения для вектора управляющих функций замкнутых систем различного типа обеспечивают движение многомерного подвижного объекта по оптимальным траекториям. Результаты реализации предложенного подхода для синтеза систем управления широким классом динамических объектов [10, 11], в том числе морских подвижных объектов, при решении задач их стабилизации в условиях ограниченных акваторий подтверждают работоспособность и эффективность применения метода структурно переключаемых обратных связей в замкнутых системах управления различного типа.

СПИСОК ЛІТЕРАТУРИ

1. *Габасов, Р.* Реализация ограниченной обратной связи в нелинейной задаче регулирования / Р. Габасов, Ф. М. Кириллова, Е. А. Ружицкая // Кибернетика и системный анализ. – 2009. – № 1 – С. 108–116.
2. *Ларин, В. Б.* Стабилизация системы обратной связью по выходной переменной / В. Б. Ларин // Проблемы управления и информатики. – 2004. – № 2. – С. 5–18.
3. *Крутько, П. Д.* Обратные задачи динамики в теории автоматического управления / П. Д. Крутько – М. : Машиностроение, 2004. – 576 с.
4. *Krutko, P. D.* Symmetry and inverse problems of control system dynamics / P. D. Krutko // Изв. РАН. Теория и системы управления. – 1996. – № 6. – С. 17–46.
5. *Крутько, П. Д.* Робастно устойчивые структуры управляемых систем высокой динамической точности. Алгоритмы и динамика управления движением модельных объектов / П. Д. Крутько // Изв. РАН. Теория и системы управления. – 2005. – № 2. – С. 120–140.
6. Справочник по теории автоматического управления / под ред. А. А. Красовского – М. : Наука, 1987. – 711 с.
7. *Тимченко, В. Л.* Оптимальное управление линейным объектом на основе метода структурно-переключаемых обратных связей // Вісник НТУ «Харківський політехнічний інститут». Тематичний випуск «Інформатика та моделювання». – 2009. – № 13. – С. 167–176.
8. *Тимченко, В. Л.* Синтез оптимальных структурно-переключаемых систем управления многомерным объектом под воздействием возмущений / В. Л. Тимченко // Технічна електродинаміка. Тем. випуск «Проблеми сучасної електротехніки». Ч. 1. – Інститут електродинаміки НАНУ, Київ. 2010. – С. 77–81.
9. *Солодов, А. В.* Линейные автоматические системы с переменными параметрами / А. В. Солодов, Ф. С. Петров. – М. : Наука, 1971. – 620 с.
10. Патент 53614 Україна МПК (2009) G05B 11/01. Спосіб керування багатовимірним динамічним об'єктом / Тимченко В. Л., Кондратенко Ю. П., Кукліна К. О. ; власник патенту Національний університет кораблебудування. – № u201004692; заявл. 20.04.2010; надрук. 11.10.2010. Бюл. № 3.
11. Патент 57100 Україна МПК (2011.01) G05B 11/01. Спосіб робастного керування багатовимірним динамічним об'єктом / Тимченко В. Л., Кондратенко Ю. П., Кукліна К. О. ; власник патенту Національний університет кораблебудування. – № u201008990; заявл. 19.07.2010; надрук. 10.02.2011. Бюл. № 19.

Надійшла 4.11.2010

Тимченко В. Л., Кондратенко Ю. П.

СИНТЕЗ СТРУКТУРНО ПЕРЕМІКАЛЬНИХ СИСТЕМ ДЛЯ КЕРУВАННЯ БАГАТОВИМІРНИМИ РУХОМИМИ ОБ'ЄКТАМИ

У статті розглядається застосування методу структурно перемикальних зворотних зв'язків до синтезу замкнених систем керування багатовимірними рухомими об'єктами. Сформовано принципи побудови оптимальних траєкторій руху об'єктів і визначення моментів перемикання керуючих функцій в ланцюгах зворотних зв'язків, отримано значення вектора керувань для замкнених систем різного типу.

Ключові слова: багатовимірні рухо мі об'єкти, структурно перемикальні зворотні зв'язки, замкнені системи керування.

Tymchenko V. L., Kondratenko Y. P.

SYNTHESIS OF STRUCTURALLY COMMUTED SYSTEMS FOR MULTIDIMENSIONAL MOVING OBJECTS CONTROL

The authors apply the method of structurally commuted feedbacks for controlling functions synthesis in closed loop systems of multidimensional moving objects control. The principles of optimal object trajectory construction and determination of controlling functions switching moments in feedback circuits are formulated; the values of control vector are obtained for different types of closed loop systems.

Key words: multidimensional moving objects, structurally commuted feedbacks, closed loop control systems.

УПРАВЛІННЯ У ТЕХНІЧНИХ СИСТЕМАХ

УПРАВЛЕНИЕ В ТЕХНИЧЕСКИХ СИСТЕМАХ

CONTROL IN TECHNICAL SYSTEMS

УДК.681.513.2

Рязанцев О. І.¹, Кардашук В. С.²¹Канд. техн. наук, завідувач кафедри Східноукраїнського національного університету ім. Володимира Даля, м. Севєродонецьк²Канд. техн. наук, старший викладач Східноукраїнського національного університету ім. Володимира Даля, м. Севєродонецьк

МЕТОДИ ТА ПРОГРАМНО-ТЕХНІЧНІ ЗАСОБИ АВТОМАТИЗАЦІЇ КЕРУВАННЯ ПРОЦЕСОМ АЕРОЗОЛЬНОГО НАНОКАТАЛІЗУ

Запропоновано методи керування процесом аерозольного нанокаталізу з використанням сучасних програмно-технічних засобів. Проведено декомпозицію об'єкта керування на підсистеми. Визначено основні задачі автоматизації керування за рахунок підтримки значень змінних – температури, частоти вібрації, амплітуди коливань реактора, компенсації збурень в заданих межах.

Ключові слова: аерозольний нанокаталіз, параметри, підсистеми, динамічне програмування, програмно-технічні засоби.

ВСТУП

Проблема науково обґрунтованого підходу при проектуванні систем автоматизації керування (САК) складними хіміко-технологічними процесами, впровадження методів і алгоритмів керування технологічним процесом аерозольного нанокаталізу для знешкодження широкого спектру відходів хлорорганічного синтезу є актуальною задачею в зв'язку з високими темпами накопичення промислових відходів хімічних виробництв [1].

Актуальність питання знешкодження широкого спектру відходів хлорорганічного синтезу викликана тим фактором, що хімічні сполуки, які утворюються в результаті промислової діяльності, негативно впливають на стан екологічної безпеки і здоров'я людини.

Метою даної статті є аналіз параметрів технологічного процесу аерозольного нанокаталізу, функцій системи автоматизації, обґрунтування та вибір методів керування, оптимізації в підсистемах технологічного процесу та програмно-технічних засобів для їх реалізації.

Дослідження теоретичних основ процесів хімічної технології, широке застосування програмно-технічних засобів (ПТЗ) створюють умови для переходу на якісно нові методи аналізу та проектування системи керування складними хіміко-технологічними процесами. При цьому зменшуються витрати на створення проміжних, дослідно-конструкторських установок, збільшується надійність та екологічна безпека проектних рішень.

АНАЛІЗ ПРОЦЕСУ АЕРОЗОЛЬНОГО НАНОКАТАЛІЗУ ЯК ОБ'ЄКТА АВТОМАТИЗОВАНОГО КЕРУВАННЯ

З метою розробки методів керування здійснено дослідження кінетики процесу аерозольного нанокаталізу, що проводився в трубчатому безградієнтному каталітичному віброреакторі в газовій фазі. Нагрів реакційного об'єму лабораторної установки проводився електричним нагрівачем (рис. 1).

Для визначення основних характеристик процесу аерозольного нанокаталізу в реактор подавався діхлоретан (ДХЕ – хімічна формула $C_2H_4Cl_2$) як один з основних компонентів відходів виробництва вінілхлориду [2].

При проведенні експериментів проводились дослідження впливу температури, концентрації каталізатора, частоти коливань реактора на процес роз-

кладу ДХЕ. З метою більш глибокого розуміння процесів, що проходять в реакторі, був проведений термодинамічний аналіз основних реакцій, що дозволив виявити умови ефективного здійснення даного процесу і якісно зв'язати весь хлор сировини в хлористий водень.

Встановлено, що температура, співвідношення реагентів, концентрація каталізатора, значення вібрації дозволяють керувати кінетикою процесу. В результаті досліджень отримано залежність швидкості реакції розкладу ДХЕ від частоти коливань реактора (рис. 2).

Дослідження процесу аерозольного нанокаталізу показали, що з перебігом часу склад вихідних газів не змінюється. Це свідчить про те, що процес аерозольного нанокаталізу стійкий з перебігом часу і активність каталізатора не зменшується, а також дозволили визначити значення константи швидкості

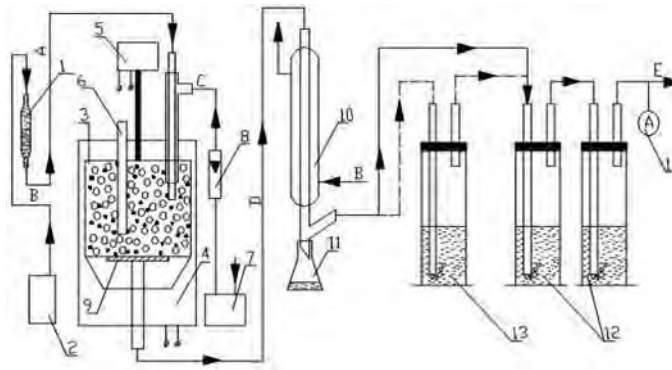


Рис. 1. Структурна схема лабораторної установки аерозольного нанокаталізу:

1 – воронка; 2 – дозатор; 3 – реактор; 4 – електрична пічка; 5 – магнітний пристрій; 6 – карман для термомпари; 7 – компресор; 8 – ротаметр; 9 – фільтр; 10 – водяний холодильник; 11 – приймач конденсату; 12 – барботери; 13 – барботер для відбору проб на HCl і Cl_2 ; 14 – точка відбору газових проб, А – 1,2-діхлоретан; В – вода; С – повітря; D – продукти реакції; Е – в атмосферу

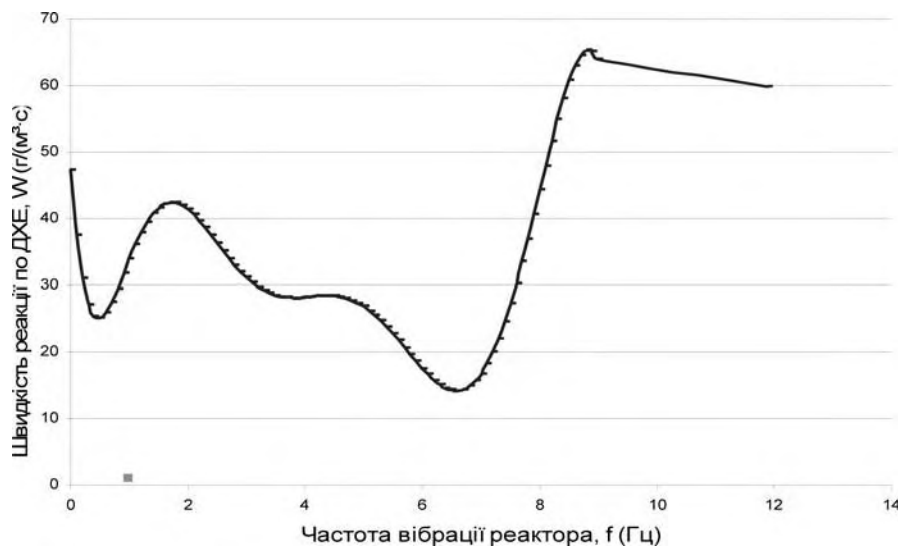


Рис. 2. Залежність швидкості реакції від частоти вібрації реактора

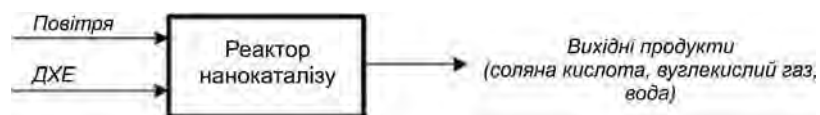


Рис. 3. Матеріальні потоки процесу аерозольного нанокаталізу

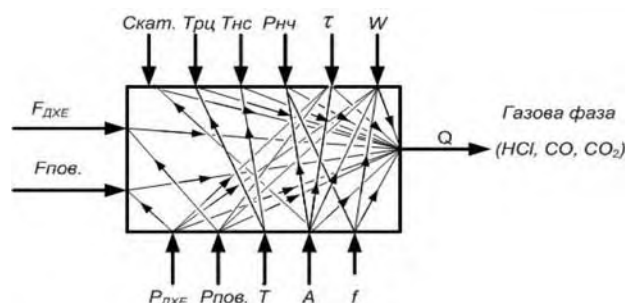


Рис. 4. Параметри процесу аерозольного нанокаталізу

і енергію активації (E) при концентрації каталізатора $2,4 \text{ г/м}^3_{\text{р.об}}$. Значення енергії активації склало $71,2 \text{ кДж/моль}$. При таких параметрах процес протікає в кінетичній області. Важливо, що в продуктах реакції не виявлено молекулярного хлору.

Екологічна безпека тісно пов'язана з безпекою функціонування хімічно небезпечних об'єктів і залежить від багатьох факторів: надійності систем автоматизації, протиаварійного захисту та ін. Огляд літератури показав, що автоматизація процесів нанохімічних технологій знаходиться на початковому етапі свого розвитку.

Аналіз процесу аерозольного нанокаталізу дозволив провести структурну декомпозицію матеріальних потоків (рис. 3).

Виходячи з рівняння матеріального балансу хімічної реакції, що проходить в реакторі нанокаталізу, розраховано питому вагу кожного компонента в процентному відношенні, яка складає для вхідних продуктів – ДХЕ – 55 %, кисню – 45 %, вихідних продуктів – вуглекислого газу – 49 %, соляної кислоти – 40 %, води – 11 %.

Протікання реакторних процесів характеризується швидкістю реакції (W) і ступенем перетворення початкової сировини в продукти реакції (x_a). На ці показники впливають концентрація реагентів, температура всередині реактора, активність каталітичної системи (каталізатор і скляні кульки), вібрація та ін. Метою керування є одержання заданого складу вихідних продуктів шляхом впливу на ці параметри.

З метою повного використання реагентів в реакторі підтримувалось задане співвідношення їх витрат. При незначній зміні завантаження реактора за допо-

могою САК забезпечувались витрати реагентів, що подаються в реактор.

Як підкреслено в [2], швидкість гетерогенних реакцій пропорційна величині поверхні дотику реагентів. Виходячи з цього, можна зробити висновок, що вібрація в процесі аерозольного нанокаталізу відіграє ключову роль і є предметом подальших досліджень.

Вплив частоти вібрації як пріоритетної змінної керування на швидкість процесу аерозольного нанокаталізу пояснюється зміною активності частинок каталізатора та реагентів. Вібраційним впливом вирішуються дві задачі процесу: активація поверхні та запобігання спіканню каталізатора. Фактично при зміні частоти коливань з'являються нові властивості поверхні каталізатора та активності наночастинок [3].

Таким чином, побічний вплив на розмір наночастинок здійснюється через температуру, частоту, амплітуду вібрації та співвідношення реагентів. Вплив факторів збурення – концентрації каталізатора, розміру наночастинок – є предметом подальших досліджень в області нанокаталізу та нанотехнологій.

В роботі [4] запропоновано метод вирішення задачі контролю за станом об'єкта в умовах невизначеності щодо запасів каталізатора за допомогою теорії нечітких множин та обробки нечіткої інформації в системах прийняття рішень.

На основі проведеного аналізу зроблено висновок, що розглянутий процес є складним багатозв'язним об'єктом керування. Виділено основні групи параметрів, що визначають хід процесу та характеризують його стан (рис. 4) [5].

Проведений аналіз дозволив виділити такі групи параметрів:

а) вхідні параметри – вхідний потік ДХЕ ($F_{\text{ДХЕ}}$), вхідний потік повітря ($F_{\text{пов.}}$);

б) змінні параметри керування – витрати ДХЕ ($P_{\text{ДХЕ}}$), витрати повітря ($P_{\text{пов.}}$), температура (T), частота вібрації (f), амплітуда (A);

в) параметри збурення – концентрація каталізатора ($C_{\text{кат.}}$), температура реакції ($T_{\text{рц.}}$), температура навколишнього середовища ($T_{\text{нс.}}$), розмір наночастинок ($P_{\text{нч.}}$), час контакту (τ), швидкість реакції (W);

г) вихідні параметри – ступінь перетворення x_a , вміст у вихідних продуктах вуглекислого газу (CO_2), чадного газу (CO), соляної кислоти (HCl), хлору (Cl_2).

Таким чином, для якісного керування процесом аерозольного нанокаталізу для параметрів $F_{\text{ДХЕ}}$ і $F_{\text{пов.}}$ необхідно підтримувати співвідношення ДХЕ до кисню як 1 до 2,5 ($\alpha = 1$) при температурі в реакторі 600 °С, частоті вібрації 10–12 Гц та амплітуді $8 \cdot 10^{-3}$ м. Об'єм реактора під час проведення експериментів складав 0,0042 м³.

При проведенні експериментів встановлено, що для незворотності процесу коефіцієнт α в подальших розрахунках необхідно прийняти як 1,02.

Отримані під час проведення експериментів значення використані як значення уставок на верхньому рівні автоматизації [6].

ФУНКЦІ СИСТЕМИ АВТОМАТИЗАЦІЇ ТА ПРОГРАМНО-ТЕХНІЧНИХ ЗАСОБІВ ДЛЯ ЇХ РЕАЛІЗАЦІЇ

Для реалізації задач автоматизації керування з виконанням необхідних вимог (якість керування, точність та ін.) необхідні відповідні ПТЗ, що застосовуються в САК.

САК процесом аерозольного нанокаталізу представлено як програмно-технічний комплекс (ПТК), що об'єднує вимірювальні датчики, виконавчі пристрої,

механізми, програмно-логічний контролер (ПЛК) нижнього рівня ієрархії з робочою станцією верхнього рівня і далі з рівнем керування підприємством. Метою функціонування САК є забезпечення ефективного керування шляхом відповідного вибору керуючих дій.

Для процесу аерозольного нанокаталізу визначено вхідні впливи, встановлено залежності між вхідними та вихідними параметрами об'єкта керування, реалізовано автоматичну зміну вхідних параметрів і керування.

Найбільш поширеною в САК є задача стабілізації, коли потрібно з заданою точністю підтримувати постійними визначені параметри процесу. Для рішення цієї задачі розроблено математичну модель процесу аерозольного нанокаталізу [5].

Задача автоматизації керування процесом аерозольного нанокаталізу вирішена таким чином: за отриманими даними про стан об'єкта керування реалізована така послідовність керуючих впливів, що функціонування САК забезпечило задовільну якість процесу за узагальненим критерієм – відсутності молекулярного хлору у вихідних продуктах (рис. 5).

Реалізація функцій верхнього рівня в САК забезпечується за допомогою робочої станції. Робоча станція займається обробкою, представленням технологічної інформації, її архівуванням, аналізом, а також виконує статистичні, мережеві функції, веде моніторинг стану ПТЗ [6].

Широке застосування в системах автоматизації технологічних процесів знайшли робочі станції фірми Advantech, які виконані на базі відмовостійких платформ серії IPC-610, IPC-615 і забезпечують ступінь захисту IP65.

На основі таких робочих станцій будуються автоматизовані робочі місця операторів-технологів. Розгалужену мережу каналів передачі інформації в системах автоматизації технологічних процесів забезпечують мережеві контролери Ethernet, RS-485 та багатоканальні модулі вводу-виводу інтерфейсу RS-232.



Рис. 5. Загальна схема системи керування процесом аерозольного нанокаталізу

В робочі станції за допомогою мережевих пристроїв зв'язку надходить технологічна інформація від ПЛК нижнього рівня, яка вже пройшла первинну обробку і яка служить основою для представлення реального стану процесу на мнемосхемах, реєстрації історії, оптимізаційних розрахунків.

Як операційна система обрана операційна система реального часу QNX фірми QSSL (QNX Software System Ltd.).

Реалізацію функцій нижнього рівня забезпечує ПЛК, який входить до складу технологічної мережі і виконує функції вводу інформації з аналогових і дискретних вхідних каналів, первинну обробку прийнятої інформації (фільтрацію, лінеаризацію, масштабування, перетворення в фізичні величини сигналів, контроль на достовірність, реєстрацію подій), функції керування (цифрове ПД-регулювання, керування виконавчими пристроями, інтелектуальне керування), функції технологічного захисту і блокування.

Гнучкість ПЛК полягає в можливості застосування під конкретні задачі широкої номенклатури аналогово-цифрових перетворювачів, модулів формування дискретних сигналів (МФДС), модулів вводу дискретних сигналів (МВДС) та ін. [7].

Для накопичення і подальшого використання оперативним персоналом інформації про перебіг процесу і дії персоналу передбачена функція архівації. Архів заповнюється при виникненні зміни стану процесу, а також через певні проміжки часу для фіксації параметрів.

Важливий критерій, який намагаються мінімізувати при обробці сигналів нижнього рівня – додаткова похибка, що вноситься при обробці аналогових сигналів, яка в даній САК не перевищує 0,1–0,15 %.

При проектуванні і розробці САК разом з правильним вибором ПТЗ, мережевого обладнання, програмних протоколів керування мережею велике значення має і розподілення програмних модулів між робочою станцією та ПЛК. При коректному розпо-

діленні задач можна збільшити продуктивність САК, зменшити час реакції на події.

МЕТОДИ МОДЕЛЮВАННЯ ТА ОПТИМІЗАЦІЇ ПРОЦЕСУ АЕРОЗОЛЬНОГО НАНОКАТАЛІЗУ

Особливістю сучасних процесів хімічної технології є велика швидкість протікання реакцій при високих температурах та їх складність, що насамперед полягає у нелінійній залежності між вхідними і вихідними параметрами. При швидкому перебігу процесу ефективна САК є вирішенням проблеми досягнення необхідної швидкодії процесів керування.

Процес аерозольного нанокаталізу належить до маловивчених і складних, що визначило експериментально-аналітичний метод дослідження з проведенням активного експерименту для розробки методів керування.

Моделювання як метод дослідження є важливим етапом при проектуванні САК (рис. 6).

На основі розробленої математичної моделі виконано аналіз технологічних підсистем об'єкта керування [5].

При аналізі спеціальної літератури не виявлено математичного опису процесу аерозольного нанокаталізу, тому в подальшому для його опису використані рівняння, що описують кінетику реактора змішування та рівняння, що описують функціональні підсистеми процесу аерозольного нанокаталізу [5].

Згідно з функціональним призначенням були визначені такі основні підсистеми процесу аерозольного нанокаталізу:

- а) нагріву;
- б) змішування реагентів;
- в) вібрації.

Слід відзначити необхідність використання адаптації в підсистемах процесу аерозольного нанокаталізу, що викликано залежністю змінних від стану процесу.



Рис. 6. Структура схема математичного моделювання

Для реалізації запропонованого методу створюються канали керування підсистемами, за допомогою яких здійснюється вплив на об'єкт керування

$$Y = F(X, U), \quad (1)$$

де F – функціональна залежність між вхідними параметрами X та вихідними параметрами Y , яка враховує керування U .

Залежність (1) є основою для створення методів керування, визначення форми її реалізації та необхідного ступеню автоматизації.

Для автоматизації керування з використанням ПЛК визначено мету Z^* керування, а також розроблено алгоритм керування ϕ , що дозволяє досягти поставленої мети на підставі поточної інформації про стан об'єкта керування та середовища:

$$U = \phi(X, Y, Z^*). \quad (2)$$

Об'єкт керування сприймає навколишнє середовище як сукупність значень деякого набору параметрів

$$S = (S_1, S_2, \dots, S_n), \quad (3)$$

кожний з яких може бути змінений:

$$S(U) = (S_1(U), S_2(U), \dots, S_n(U)). \quad (4)$$

Процес моделювання передбачає знаходження характеру залежності стану об'єкта F від його вхідних параметрів X , параметрів збурення G і керування U :

$$Y = F(X, G, U). \quad (5)$$

Експертний метод рішення є основним в структурному аналізі моделі і передбачає упорядкування вхідних і вихідних параметрів об'єкта керування.

Параметричний аналіз моделі пов'язаний з визначенням параметрів $C = (c_1, c_2, \dots, c_n)$ моделі:

$$Y = F(X, G, U, C), \quad (6)$$

де вибрана на попередньому етапі структура St відображена в модельному операторі. При такому підході, очевидно, треба мати інформацію про поведінку цих параметрів. Тут можливі два підходи: ідентифікація та планування експерименту.

Ідентифікація параметрів моделі F пов'язана з оцінкою чисельних значень параметрів в стаціонарному режимі, тобто без організації керуючих впливів на нього. Вихідною інформацією для ідентифікації є структура St та спостереження за поведінкою входу $X(t)$ та виходу $Y(t)$.

Таким чином, пара

$$J(t) = \langle X(t), Y(t) \rangle, \quad (7)$$

отримана в режимі нормального функціонування системи, є основним джерелом інформації при ідентифікації.

Для з'ясування залежності виходу $Y(t)$ від параметрів $X(t)$, $G(t)$, $U(t)$ може бути застосовано експеримент.

В САК збурення, як правило, не носять хаотичний характер. Вони або повільно змінюються в часі, або є стрибкоподібними. Якщо попередньо що-небудь сказати про характер збурення не можна, то при розробці системи розрахунок ведеться для деяких типових збурень. Одним з таких типових збурень є збурення типу «стрибок». Для визначення динамічних властивостей об'єкта керування визначається реакція величини, що регулюється, на типові збурення. Найбільш поширеним є використання кривої розгону – функції зміни параметра в часі, яка викликана разовим стрибкоподібним збуренням на вході [8].

Процес керування передбачає прийняття рішення для значення U , щоб в ситуації S , що склалась, досягнути заданої мети Z^* . Це рішення базується на використанні моделі F , мети Z^* , отриманої інформації про стан X , Y , а також на наявності ресурсів керування R , які є обмеженнями, що накладаються на U . Адаптація є частковим випадком керування і полягає в зміні фактора керування U таким чином, щоб підтримувати деякі функціонали об'єкта керування в потрібному стані, незалежно від дії всякого роду зовнішніх та внутрішніх дій. При цьому специфіка об'єкта накладає на керування вимогу

$$U \in S, \quad (8)$$

де S – множина допустимих керувань.

При аналізі методів, що використовуються для оптимізації параметрів ТП, важливу роль відіграє проблема розмірності векторів стану та керуючих впливів. Так, метод динамічного програмування дав можливість розбити задачу вибору оптимального керування всього об'єкта $U^{(i)}$ ($i = 1, 2, \dots, n$) на n задач, в кожній з яких вибирається тільки одне керування $U^{(i)}$ [9].

Адаптація для процесу аерозольного нанокаталізу виникає в тому випадку, коли відсутня інформація, необхідна для оптимізації об'єкта керування (рис. 7).

При дослідженні процесу аерозольного нанокаталізу встановлено, що ефективність кожної підсистеми оцінюється скалярною величиною

$$r_i = r_i^*(x^i, u^i), \quad (9)$$

заданою у вигляді функції від змінних стану підсистеми x^i та застосованого в цій підсистемі керування u^i .

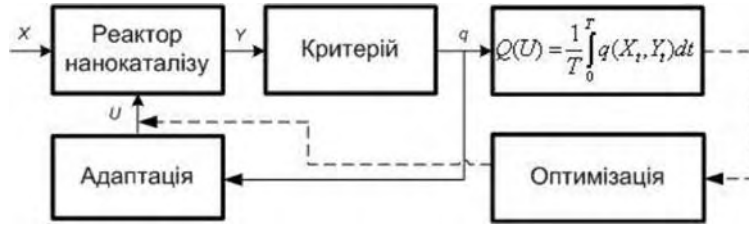


Рис. 7. Структурна схема адаптації та оптимізації

Результуюча оцінка ефективності в цілому визначена як адитивна функція результатів, отриманих від кожної підсистеми:

$$R_n = \sum_{i=1}^n r_i(x^s, u^s), \quad (10)$$

де r_i – критерій оптимальності підсистеми; x^i – технологічні параметри підсистеми; u^i – оптимальне керування технологічними параметрами підсистеми.

Таким чином, задача оптимізації сформульована як задача знаходження оптимальної стратегії:

$$\tilde{U}_{n_{\text{онт.}}} = (U_{1_{\text{онт.}}}, U_{2_{\text{онт.}}}, \dots, U_{n_{\text{онт.}}}), \quad (11)$$

яка максимізує величину R_n .

Позначимо через F_n максимальне значення критерію R_n , яке можна отримати при використанні оптимальної стратегії керування $\tilde{U}_{n_{\text{макс}}}$. Очевидно, що F_n також залежить від стану всіх підсистем процесу і може бути визначена як

$$F_n(x) = \max R_n x^i. \quad (12)$$

Процедура застосування принципу оптимальності для процесу аерозольного нанокаталізу, який складається з n підсистем, починається з останньої стадії, для котрої не існує наступних стадій, які, згідно принципу оптимальності, не можуть вплинути на вибір оптимального керування.

Для реалізації пошуку оптимальних значень змінних в підсистемах для процесу аерозольного нанокаталізу застосовано метод пошуку максимального значення функції за сіткою змінних, суть якого для функції однієї змінної полягає в тому, що значення функції $x(t)$ розраховується для n рівновіддалених значень незалежної змінної t в інтервалі її зміни $t(0) \leq t \leq t(n)$. Очевидно, що при застосуванні такого методу похибка в визначенні положення зменшується із збільшенням кількості вимірів n .

В даному методі виникає проблема визначення самої величини n . Збільшення її значення призводить до надмірних операцій обчислення, що позначається на продуктивності всієї системи, оскільки є ще інші параметри, які також треба обчислювати.

Задача ускладнюється, якщо шукається екстремум функції двох змінних $x(t_1)$ та $x(t_2)$ в області зміни параметрів, які характеризуються нерівностями

$$t_1^{(0)} \leq t_1 \leq t_1^{(n)}, \quad t_2^{(0)} \leq t_2 \leq t_2^{(n)}. \quad (13)$$

При такій постановці задачі значення функції в усіх вузлах сітки дорівнює n^2 .

Для САК процесом аерозольного нанокаталізу зазначено, що кожна підсистема оптимізує свою цільову функцію, а верхній рівень координує рішення підсистем нижнього рівня для досягнення оптимального критерію глобальної цільової функції та реалізації методу чіткої (жорсткої) координації, який полягає в тому, що для параметра, який координується, в кожен момент часу встановлюється вимога

$$K(t) = K. \quad (14)$$

ВИСНОВКИ

В результаті аналізу процесу аерозольного нанокаталізу та запропонованих методів керування отримано такі практичні результати.

1. Досліджено процес аерозольного нанокаталізу як складний багатозв'язний об'єкт керування, здійснено аналіз методів та програмно-технічних засобів реалізації задач керування. Наведено структурну схему та взаємозв'язок основних параметрів процесу аерозольного нанокаталізу.

2. Проведено декомпозицію процесу на підсистеми стабілізації температури, вібрації, амплітуди та змішування реагентів. Виділено основні задачі автоматизації керування за рахунок регулювання температури, частоти вібрації, амплітуди коливань реактора, співвідношення реагентів в заданих межах та компенсації збурень з метою забезпечення максимального питомого виходу цільового продукту та повної відсутності хлору у вихідних продуктах.

3. Відзначено доцільність використання для автоматизації керування методу динамічного програмування, що дозволяє розбити загальну задачу керування технологічним процесом на n задач, в кожній з яких визначається лише одна змінна керування.

4. Показано, що ефективність кожної підсистеми може бути оцінена скалярною величиною, заданою у вигляді функції від технологічних параметрів підсистеми та застосованого в ній керування. Загальна оцінка ефективності технологічного процесу в цілому визначена як адитивна функція результатів, отриманих від кожної підсистеми. Сформульовано задачу знаходження оптимальної стратегії керування, яка максимізує таку глобальну функцію.

5. Згідно з проведеною декомпозицією створено відповідні канали керування, за допомогою яких здійснюється вплив на стан технологічного процесу.

6. Визначено мету та алгоритм функціонування програмно-логічного контролера, що функціонує в складі загальної системи автоматизації для забезпечення ефективного керування на основі інформації про стан процесу аерозольного нанокаталізу.

СПИСОК ЛІТЕРАТУРИ

1. *Рязанцев, А. И.* Применение современных технологий при проектировании многоканальных блоков ввода-вывода для систем экологического мониторинга / А. И. Рязанцев, В. С. Кардашук // Системы контроля и управления технологическими процессами : сб. науч. статей ; под общей ред. В. В. Елисеева. – Луганск : Світлиця, 2006. – С. 258–265.
2. *Гликин, М. А.* Оксидегидрохлорирование отходов. Аэрозольный нанокаталіз с утилизацией хлора / М. А. Гликин, Л. А. Баранова, И. Н. Сологуб // Хімічна промисловість України. – 2006. – № 2. – С. 19–25.
3. *Рязанцев, О. І.* Дослідження та застосування методу керування підсистемою вібрації для процесу аерозольного нанокаталізу / О. І. Рязанцев, В. С. Кардашук // Вісник Східноукраїнського національного університету імені Володимира Даля. – 2010. – № 9(151). – С. 184–189.
4. *Рязанцев, О. І.* Керування процесом аерозольного нанокаталізу в умовах невизначеності по запасам каталізатора / О. І. Рязанцев, В. С. Кардашук // Вісник Східноукраїнського національного університету імені Володимира Даля. – 2010. – № 6(148). – Ч. 2. – С. 94–98.
5. *Рязанцев, О. І.* Побудова математичної моделі технологічного процесу аерозольного нанокаталізу у віброзрідженному шарі для організації керування / О. І. Рязанцев, В. С. Кардашук // Вісник Східноукраїнського національного університету імені Володимира Даля. – 2009. – № 6(136). – Ч. 1. – С. 274–279.
6. *Рязанцев, О. І.* Система автоматизації керування технологічним процесом аерозольного нанокаталізу у віброзрідженному шарі / О. І. Рязанцев, В. С. Карда-

шук // Радіоелектроніка, інформатика, управління. – 2010. – № 1(22). – С. 164–172.

7. *Рязанцев, А. И.* Система управления на базе современных программно-технических средств процессами утилизации отходов химических производств по технологии нанокаталіза / А. И. Рязанцев, В. С. Кардашук // Вісник Східноукраїнського національного університету імені Володимира Даля. – 2008. – № 8(126). – Ч. 1. – С. 318–326.
8. *Кардашук, В. С.* Методи керування підсистемою температури для процесу аерозольного нанокаталізу в системах утилізації відходів хлорорганічного синтезу / Кардашук В. С., Рязанцев О. І. // Технологія-2010 : XIII Всеукраїнська наук.-практ. конф. студентів, аспірантів та молодих вчених з міжнар. участю, 22–23 квітня 2010 р. : матеріали. – Северодонецьк, 2010. – Ч. 1. – С. 97–99.
9. *Рязанцев, О. І.* Застосування методу динамічного програмування для реалізації задачі оптимізації технологічного процесу аерозольного нанокаталізу у віброзрідженному шарі / О. І. Рязанцев, В. С. Кардашук // Вісник Східноукраїнського національного університету імені Володимира Даля. – 2009. – № 12(142). – Ч. 2. – С. 109–116.

Надійшла 16.11.2010

Рязанцев А. И., Кардашук В. С.

МЕТОДЫ И ПРОГРАММНО-ТЕХНИЧЕСКИЕ СРЕДСТВА АВТОМАТИЗАЦИИ УПРАВЛЕНИЯ ПРОЦЕССОМ АЭРОЗОЛЬНОГО НАНОКАТАЛИЗА

Предложены методы управления процессом аэрозольного нанокаталіза с применением современных программно-технических средств. Проведена декомпозиция объекта управления на подсистемы. Определены основные задачи автоматизации управления за счет поддержания значений переменных – температуры, частоты вибрации, амплитуды колебаний реактора, компенсации возмущений – в заданных пределах.

Ключевые слова: аэрозольный нанокаталіз, параметры, подсистемы, динамическое программирование, программно-технические средства.

Ryazantsev A. I., Kardashuk V. S.

METHODS AND PROGRAM-TECHNICAL MEANS FOR AEROSOL NANOCATALYSIS AUTOMATIC CONTROL

The authors propose the method of aerosol nanocatalysis control using modern program-technical means. The object of control is resolved into subsystems. The main tasks of control automation are realized by maintaining the values of variables – temperature, vibration frequency, reactor vibration amplitude, disturbances compensation – within the specified limits.

Key words: aerosol nanocatalysis, parameters, subsystems, dynamic programming, program-technical means.

ДО ВІДОМА АВТОРІВ

Журнал «Радіоелектроніка, інформатика, управління» (PIU) призначений для публікації найбільш значимих наукових і практичних результатів досліджень учених вищих навчальних закладів і наукових організацій.

Журнал включений у перелік наукових видань України, у яких можуть публікуватися результати дисертаційних робіт на змагання вчених ступенів доктора і кандидата технічних наук і фізико-математичних наук (фізика).

Статті, що опубліковано в журналі, реферуються в реферативних журналах і базах даних ВІНІТІ (Росія) і «Джерело» (Україна). Інтернет-сторінка журналу:

<http://journal.zntu.edu.ua/tic/index.php?page=index>.

Журнал видається два рази в рік і розповсюджується по підписці в Україні. Передплатний індекс 22914.

Для розгляду питання про публікацію статті в редакцію журналу необхідно вислати поштою або представити особисто наступне:

1) рукопис (роздруківку) статті, підписаний на останній сторінці всіма авторами, в двох екземплярах;

2) оригінал експертного висновку про можливість відкритого опублікування статті;

3) супровідний лист-клопотання з організації, де була виконана робота (або лист автора);

4) рецензію від фахівця в даній області з вченим ступенем доктора наук. Підпис рецензента обов'язково мусить бути завіреним.

5) диск з наступними файлами:

1) електронна версія статті, повністю ідентична роздруківці;

2) відомості про авторів;

3) рисунки у графічному форматі .tif.

Файли з матеріалами статті можна вислати електронною поштою або передати особисто на оптичному диску або USB-накопичувачі.

Вимоги до оформлення статті. Приймаються статті, набрані в редакторі Microsoft Word.

Параметри сторінки:

– розмір паперу – А4 (210x297);

– орієнтація – книжкова;

– шрифт – Times New Roman, розмір – 12 pt;

– міжрядковий інтервал – полуторний;

– верхнє поле – 20 мм, нижнє – 20 мм, ліве – 25 мм, праве – 15 мм.

Сторінки рукопису повинні бути пронумеровані. Не допускаються розбіжності рукопису з електронною версією статті. Текст рукопису не повинен мати рукописних виправлень та позначок.

Послідовність розміщення матеріалу статті:

1) індекс УДК;

2) прізвища й ініціали авторів, назва статті, анотація й ключові слова українською мовою (для громадян України);

3) прізвища й ініціали авторів, назва статті, анотація й ключові слова російською мовою;

4) прізвища й ініціали авторів, назва статті, анотація й ключові слова англійською мовою;

5) текст статті;

6) список літератури.

Текст статті. Приймаються статті російською, українською та англійською мовами. Розмір статті до 0,5 авторського аркуша. У статті слід уникати зайвої деталізації, про-

можливих формул і висновків; не слід наводити відомі факти, повторювати зміст таблиць і ілюстрацій у тексті. Стаття не повинна мати граматичних або інших помилок, а також повинна відповідати тематиці журналу й вимогам ВАК щодо фахових видань. **Структура** тексту статті мусить містити такі необхідні елементи: постановка проблеми в загальному виді і її зв'язок з важливими науковими або практичними завданнями; аналіз останніх досліджень і публікацій, у яких розпочато розв'язання даної проблеми, і на які опирається автор; виділення нерозв'язаних раніше частин загальної проблеми, яким присвячується стаття; формулювання цілей статті (постановка завдання); виклад основного матеріалу дослідження з повним обґрунтуванням отриманих наукових результатів, висновки по даному дослідженню й перспективи подальших досліджень у даному напрямку. Матеріал публікації мусить бути розбитий на підрозділи не більше двох рівнів.

Рисунки розміщуються в тексті й додатково додаються в окремих файлах (формат .tif з роздільною здатністю 150–300 dpi, чорно-білі або у градаціях сірого). Розмір рисунків не повинен перевищувати ширини сторінки (17 см) або ширини колонки (8 см). Написи на рисунках бажано виконувати шрифтом Times New Roman, розмір 10. Рисунки нумерують і підписують унизу.

Формули виконуються за допомогою вбудованого в Word редактора Microsoft Equation. Формули нумерують у круглих дужках праворуч. Формули великого розміру записуються в кілька рядків.

Нумерація рисунків, формул і таблиць наскрізна однакова.

Обсяг **анотації** не повинен перевищувати 40 слів.

Ключові слова наводяться в називному відмінку у кількості до десяти слів.

Список літератури наприкінці статті подається мовою оригіналу і складається в порядку згадування посилань у тексті й відповідно до діючого стандарту на бібліографічний опис. Посилання на літературу в тексті нумеруються послідовно й позначаються цифрою у квадратних дужках.

У **відомостях про авторів** необхідно навести:

1) прізвище, ім'я, по батькові (повністю);

2) учений ступінь;

3) посаду;

4) місце роботи;

5) робочу чи домашню і електронну адресу;

6) робочий, домашній, мобільний телефон.

Статті, які не відповідають зазначеним вимогам, не приймаються до розгляду.

Всі статті проходять закриті рецензування і в разі потреби можуть бути повернуті автору на доробку. Редакція залишає за собою право на літературну редакцію тексту статті без повідомлення автору.

Рукописи й диски не вертаються, коректура та відбитки статей авторам не надсилаються.

Адреса редакції: 69063, м. Запоріжжя, вул. Жуковського, 64, ЗНТУ, редакція журналу «PIU»

Тел.: (061) 7-698-2-96 – редакційно-видавничий відділ

(061) 7-644-6-62 – головний редактор

Факс: (061) 7-642-1-41

E-mail: rvv@zntu.edu.ua

Наукове видання

**Радіoeлектроніка,
інформатика,
управління**

№ 1/2011

Науковий журнал

Головний редактор – д-р техн. наук Піза Д. М.

Заст. головного редактора – канд. техн. наук Дубровін В. І.

Комп'ютерне моделювання та верстання

Новікова К. О.

Оригінал-макет підготовлено у редакційно-видавничому відділі ЗНТУ

*Підписано до друку 30.06.2011. Формат 60x84/8.
Папір офс. Різогр. друк. Ум. друк. арк. 20,2.
Тираж 300 прим. Зам. № 1182.*

69063 м. Запоріжжя, ЗНТУ, друкарня, вул. Жуковського, 64