

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Том 23

2017

№ 7

ТЕОРЕТИЧЕСКИЙ И ПРИКЛАДНОЙ НАУЧНО-ТЕХНИЧЕСКИЙ ЖУРНАЛ

Издается с ноября 1995 г.

УЧРЕДИТЕЛЬ

Издательство "Новые технологии"

СОДЕРЖАНИЕ

ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ

- Астафуров В. Г., Евсюткин Т. В., Аксёнов С. В. Эффективный алгоритм формирования нечетких систем на основе модели Питтсбургского классификатора. . . 483
- Волобуев М. Ф., Мальцев А. М., Соколов И. В. Экспериментальные исследования макета резервированной системы, управляемой нечетким контроллером. . . 491

МОДЕЛИРОВАНИЕ И ОПТИМИЗАЦИЯ

- Левин В. И. Многомерные интервалы, их исчисление и применение. 499
- Титов Д. В., Ретинская И. В., Ретинский В. С. Моделирование инфраструктуры центра сертификации 503
- Литвин В. Г., Литвин Ю. В. Об оценке времени разработки крупномасштабной информационно-управляющей системы, создаваемой в структуре системы систем 509

БЕЗОПАСНОСТЬ ИНФОРМАЦИИ

- Алгулиев Р. М., Имамвердиев Я. Н., Сухостат Л. В. Киберфизические системы: основные понятия и вопросы обеспечения безопасности 517

ГЕОИНФОРМАЦИОННЫЕ СИСТЕМЫ

- Ружичка З. З., Музалевский К. В., Якубайлик О. Э., Швецов Е. Г. Информационно-технологическое обеспечение задач обработки и представления радиометрических спутниковых данных 529

СИСТЕМЫ АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ

- Печенко И. С., Петров А. Б. Модели данных для архитектурных спецификаций систем на кристалле 536

НЕЙРОСЕТЕВЫЕ ТЕХНОЛОГИИ

- Гейдаров П. Ш. Нейронные сети прямого распространения с вычисляемыми параметрами 543

Главный редактор:

СТЕМПКОВСКИЙ А. Л.,
акад. РАН, д. т. н., проф.

Зам. главного редактора:

ИВАННИКОВ А. Д., д. т. н., проф.
ФИЛИМОНОВ Н. Б., д. т. н., с.н.с.

Редакционный совет:

БЫЧКОВ И. В., акад. РАН, д. т. н.
ЖУРАВЛЕВ Ю. И.,
акад. РАН, д. ф.-м. н., проф.
КУЛЕШОВ А. П.,
акад. РАН, д. т. н., проф.
ПОПКОВ Ю. С.,
акад. РАН, д. т. н., проф.
РУСАКОВ С. Г.,
чл.-корр. РАН, д. т. н., проф.
РЯБОВ Г. Г.,
чл.-корр. РАН, д. т. н., проф.
СОЙФЕР В. А.,
акад. РАН, д. т. н., проф.
СОКОЛОВ И. А., акад.
РАН, д. т. н., проф.
СУЕТИН Н. В., д. ф.-м. н., проф.
ЧАПЛЫГИН Ю. А.,
акад. РАН, д. т. н., проф.
ШАХНОВ В. А.,
чл.-корр. РАН, д. т. н., проф.
ШОКИН Ю. И.,
акад. РАН, д. т. н., проф.
ЮСУПОВ Р. М.,
чл.-корр. РАН, д. т. н., проф.

Редакционная коллегия:

АВДОШИН С. М., к. т. н., доц.
АНТОНОВ Б. И.
БАРСКИЙ А. Б., д. т. н., проф.
ВАСЕНИН В. А., д. ф.-м. н., проф.
ВИШНЕКОВ А. В., д. т. н., проф.
ДИМИТРИЕНКО Ю. И., д. ф.-м. н., проф.
ДОМРАЧЕВ В. Г., д. т. н., проф.
ЗАБОРОВСКИЙ В. С., д. т. н., проф.
ЗАГИДУЛЛИН Р. Ш., к. т. н., доц.
ЗАРУБИН В. С., д. т. н., проф.
КАРПЕНКО А. П., д. ф.-м. н., проф.
КОЛИН К. К., д. т. н., проф.
КУЛАГИН В. П., д. т. н., проф.
КУРЕЙЧИК В. В., д. т. н., проф.
ЛЬВОВИЧ Я. Е., д. т. н., проф.
МАРТЫНОВ В. В., д. т. н., проф.
МИХАЙЛОВ Б. М., д. т. н., проф.
НЕЧАЕВ В. В., к. т. н., проф.
ПОЛЕЩУК О. М., д. т. н., проф.
САКСОНОВ Е. А., д. т. н., проф.
СОКОЛОВ Б. В., д. т. н., проф.
ТИМОНИНА Е. Е., д. т. н., проф.
УСКОВ В. Л., к. т. н. (США)
ФОМИЧЕВ В. А., д. т. н., проф.
ШИЛОВ В. В., к. т. н., доц.

Редакция:

БЕЗМЕНОВА М. Ю.
ГРИГОРИН-РЯБОВА Е. В.
ЛЫСЕНКО А. В.
ЧУГУНОВА А. В.

Информация о журнале доступна по сети Internet по адресу <http://novtex.ru/IT>.

Журнал включен в систему Российского индекса научного цитирования и базу данных RSCI на платформе Web of Science.

Журнал входит в Перечень научных журналов, в которых по рекомендации ВАК РФ должны быть опубликованы научные результаты диссертаций на соискание ученой степени доктора и кандидата наук.

INFORMATION TECHNOLOGIES

INFORMACIONNYYE TEHNOLOGII

Vol. 23
2017
No. 7

THEORETICAL AND APPLIED SCIENTIFIC AND TECHNICAL JOURNAL

Published since November 1995

ISSN 1684-6400

CONTENTS

INTELLIGENT SYSTEMS AND TECHNOLOGIES

- Astafurov V. G., Evsjutkin T. V., Axyonov S. V.** An Effective Algorithm of Fuzzy System Design Based on Pittsburgh Classifier Model 483
- Volobuev M. F., Maltsev A. M., Sokolov I. V.** An Experimental Study of the Layout of a Redundant System Controlled by Fuzzy Controller 491

MODELING AND OPTIMIZATION

- Levin V. I.** Multidimensional Intervals: Calculs and Applications. 499
- Titov D. V., Retinskaya I. V., Retinsky V. S.** Modeling of the Certification Center Infrastructure 503
- Litvin V. G., Litvin Yu. V.** About Assessment of Time Development Large-Scale Management Information System, Created in Structure System of Systems. 509

CRYPTOSAFETY INFORMATION

- Alguliyev R. M., Imamverdiyev Ya. N., Sukhostat L. V.** Cyber-Physical Systems: Basic Concepts and Security Issues 517

GEOINFORMATION SYSTEMS

- Ruzicka Z. Z., Muzalevskiy K. V., Yakubailik O. E., Shvetsov E. G.** Information Technological Support of Tasks of Processing and Submission of Radiometric Satellite Data 529

CAD-SYSTEMS

- Pechenko I. S., Petrov A. B.** Data Models for SoC Specifications Creation and Usage 536

NEUROTECHNOLOGIES

- Geidarov P. Sh.** The Feedforward Neural Network with Calculated Parameters 543

Editor-in-Chief:

Stempkovsky A. L., Member of RAS,
Dr. Sci. (Tech.), Prof.

Deputy Editor-in-Chief:

Ivannikov A. D., Dr. Sci. (Tech.), Prof.
Filimonov N. B., Dr. Sci. (Tech.), Prof.

Chairman:

Bychkov I. V., Member of RAS,
Dr. Sci. (Tech.), Prof.
Zhuravljov Yu. I., Member of RAS,
Dr. Sci. (Phys.-Math.), Prof.
Kuleshov A. P., Member of RAS,
Dr. Sci. (Tech.), Prof.
Popkov Yu. S., Member of RAS,
Dr. Sci. (Tech.), Prof.
Rusakov S. G., Corresp. Member of RAS,
Dr. Sci. (Tech.), Prof.
Ryabov G. G., Corresp. Member of RAS,
Dr. Sci. (Tech.), Prof.
Soifer V. A., Member of RAS,
Dr. Sci. (Tech.), Prof.
Sokolov I. A., Member of RAS,
Dr. Sci. (Phys.-Math.), Prof.
Suetin N. V.,
Dr. Sci. (Phys.-Math.), Prof.
Chaplygin Yu. A., Member of RAS,
Dr. Sci. (Tech.), Prof.
Shakhnov V. A., Corresp. Member of RAS,
Dr. Sci. (Tech.), Prof.
Shokin Yu. I., Member of RAS,
Dr. Sci. (Tech.), Prof.
Yusupov R. M., Corresp. Member of RAS,
Dr. Sci. (Tech.), Prof.

Editorial Board Members:

Avdoshin S. M., Cand. Sci. (Tech.), Ass. Prof.
Antonov B. I.
Barsky A. B., Dr. Sci. (Tech.), Prof.
Vasenin V. A., Dr. Sci. (Phys.-Math.), Prof.
Vishnekov A. V., Dr. Sci. (Tech.), Prof.
Dimitrienko Yu. I., Dr. Sci. (Phys.-Math.), Prof.
Domrachev V. G., Dr. Sci. (Tech.), Prof.
Zaborovsky V. S., Dr. Sci. (Tech.), Prof.
Zagidullin R. Sh., Cand. Sci. (Tech.), Ass. Prof.
Zarubin V. S., Dr. Sci. (Tech.), Prof.
Karpenko A. P., Dr. Sci. (Phys.-Math.), Prof.
Kolin K. K., Dr. Sci. (Tech.)
Kulagin V. P., Dr. Sci. (Tech.), Prof.
Kureichik V. V., Dr. Sci. (Tech.), Prof.
Ljvovich Ya. E., Dr. Sci. (Tech.), Prof.
Martynov V. V., Dr. Sci. (Tech.), Prof.
Mikhailov B. M., Dr. Sci. (Tech.), Prof.
Nechaev V. V., Cand. Sci. (Tech.), Ass. Prof.
Poleschuk O. M., Dr. Sci. (Tech.), Prof.
Saksonov E. A., Dr. Sci. (Tech.), Prof.
Sokolov B. V., Dr. Sci. (Tech.)
Timonina E. E., Dr. Sci. (Tech.), Prof.
Uskov V. L. (USA), Dr. Sci. (Tech.)
Fomichev V. A., Dr. Sci. (Tech.), Prof.
Shilov V. V., Cand. Sci. (Tech.), Ass. Prof.

Editors:

Bezmenova M. Yu.
Grigorin-Ryabova E. V.
Lysenko A. V.
Chugunova A. V.

Complete Internet version of the journal at site: <http://novtex.ru/IT>.

According to the decision of the Higher Certifying Commission of the Ministry of Education of Russian Federation, the journal is inscribed in "The List of the Leading Scientific Journals and Editions wherein Main Scientific Results of Theses for Doctor's or Candidate's Degrees Should Be Published"

УДК 004.89:004.93:528.88

В. Г. Астафуров^{1, 2}, д-р физ.-мат. наук, ст. науч. сотр., e-mail: astafurov@iao.ru,
Т. В. Евсюткин¹, канд. техн. наук, мл. науч. сотр., e-mail: oomphi.iao@gmail.com,
С. В. Аксёнов^{2, 3, 4}, канд. техн. наук, доц., e-mail: axoenows@tpu.ru

¹Институт оптики атмосферы им. В. Е. Зуева СО РАН, г. Томск

²Томский университет систем управления и радиоэлектроники, г. Томск

³Национальный исследовательский Томский политехнический университет, г. Томск

⁴Национальный исследовательский Томский государственный университет, г. Томск

Эффективный алгоритм формирования нечетких систем на основе модели Питтсбургского классификатора

Предложен алгоритм автоматической классификации текстур слабоконтрастных объектов на основе модели нечеткого вывода, известной как Питтсбургский классификатор. Описаны разработанные методики построения базы правил нечетких продукций и формирования наборов информативных текстурных признаков. Представлено описание численного метода инициализации функций принадлежности и генетического алгоритма адаптации параметров нечеткой системы. Эффективность алгоритмов демонстрируется на классификации изображений четырех подтипов перистой облачности по спутниковым данным MODIS.

Ключевые слова: классификация, нечеткая система, текстурные признаки, Питтсбургский классификатор, подтипы перистой облачности

Введение

Классификация изображений является актуальной задачей при анализе объектов на спутниковых снимках, в медицине, радиолокации и ряде других областей. Основные усилия в развитии алгоритмов классификации в настоящее время направлены на повышение качества результатов. Одна из таких задач связана с проблемой распознавания слабоконтрастных объектов, представляющих собой сложноструктурированные классы, описание которых при создании систем классификации является неполным или нечетким. Для решения данной задачи можно использовать концепцию нечеткого моделирования [1], которая является эффективным средством описания поведения плохо формализованных объектов, систем и процессов и заключается во введении степени принадлежности объекта к множеству классов.

Одним из приложений автоматической классификации изображений является обработка данных дистанционного зондирования Земли (ДЗЗ) для определения типа облачности согласно метеорологическому стандарту [2]. При этом в качестве слабоконтрастных объектов выступают подтипы перистой облачности, которая из-за их малой оптической

толщины создает экран, искажающий изображение подстилающей поверхности (ПП) и нижележащей облачности.

Идентификация подтипов перистой облачности необходима в нефанализе для прогноза погоды, определения направлений ветра и зон турбулентности для обеспечения безопасности полета воздушных судов, моделирования радиационного баланса системы "атмосфера — подстилающая поверхность", а также способствует более эффективному обнаружению тепловых аномалий и очагов горения на ПП [3]. При этом задача автоматической классификации перистой облачности по подтипам в известных нам отечественных и зарубежных работах не рассмотрена.

Исходя из вышесказанного, можно сформулировать цель настоящего исследования — разработка эффективного алгоритма на базе нечеткой логики для классификации слабоконтрастных объектов на спутниковых снимках на примере четырех подтипов перистой облачности: когтевидных (*Ci unc*), хлопьевидных (*Ci floc*), послегрозовых (*Ci ing*), хребтовидных и перепутанных (*Ci vert & Ci int*).

При разработке системы автоматической классификации изображений необходимо пройти четыре

этапа: поиск и формирование исходных данных, выбор классификационных характеристик, построение классификатора и оценка его эффективности. Важнейшими этапами при создании нечеткой системы является идентификация ее структуры и параметров. Механизм нечеткого вывода в своей основе имеет базу правил, формируемую специалистами и экспертами предметной области в виде конечной совокупности продукционных правил, состоящих из лингвистических переменных [4] и отношений между ними.

Классификационные характеристики

Выбор подхода для описания изображений классов объектов зависит от решаемой задачи и типа исходных данных [5]. Наиболее достоверные и полные результаты классификации облаков по типам получены при использовании текстурной информации изображений [6, 7], позволяющей представить облачность как единую целостную структуру. Поэтому в качестве лингвистических переменных в работе используются текстурные признаки.

Для описания текстуры используют четыре метода: вектор разности уровней яркости (GLDV) [8], матрицы смежности уровней яркости (GLCM) [9], гистограммы суммы и разности (SADH) [10], а также набор признаков, учитывающих значения яркости отдельных пикселей изображения (ODSH) [11]. Исходный набор состоит из 16 наименований признаков GLCM, 8 — GLDV, 15 — SADH и 8 — ODSH, рассчитанных с учетом сдвига и углового направления между соседними элементами изображения.

Алгоритм формирования нечетких систем для решения задачи классификации

Не каждый признак или их совокупность из исходного набора является информативным с точки зрения распознавания конкретного класса объектов. Результатом предлагаемого алгоритма формирования нечетких систем также являются наборы информативных текстурных признаков, которые формируются на основе сравнительного анализа гистограмм их выборочных значений, найденных по фрагментам изображений.

На рис. 1 представлена схема алгоритма формирования нечеткой системы для решения задачи классификации изображений.

Алгоритм реализует следующие этапы.

1. Для каждого класса объектов формируется набор из N_k эталонных образцов, значения текстурных признаков (ТП) которых будут использоваться при обучении классификаторов.

2. Для каждого образца k -го класса рассчитывается набор ТП $\{T_j^k\}$, где индекс j определяет вид ТП

и соответствующий метод описания и анализа текстур изображений: GLDV [8], GLCM [9], SADH [10] и SFR [11]. Так как значения всех ТП изменяются в разных диапазонах, то они приводятся к интервалу $[0, 1]$ линейным масштабированием $\tilde{T} = (T - T_{\min}) / (T_{\max} - T_{\min})$, где T — текущее значение признака, \tilde{T} — нормированное значение ТП, T_{\max} и T_{\min} — его максимальное и минимальное значения на рассматриваемых фрагментах.

3. Для каждого \tilde{T}_j^k вычисляются гистограммы по набору рассчитанных значений $\{t_{ji}^k\}$ ТП для N_k эталонных образцов k -го класса, $i = 1, \dots, N_k$. Относительные частоты $\tilde{P}_l^k(j) = v_l^k(j) / N_k$ являются оценками вероятностей попадания ТП \tilde{T}_j^k в l -й интервал группировки гистограммы d_l , где $l = 1, \dots, L$, L — число интервалов группировки, $v_l^k(j)$ — число значений j -го ТП, принадлежащих интервалу d_l ,

$$N_k = \sum_{l=1}^L v_l^k(j).$$

4. Фиксируется конкретный класс c_k , который требуется распознать на изображениях, а остальные классы c_i считаются "фоновыми", $i \neq k$.

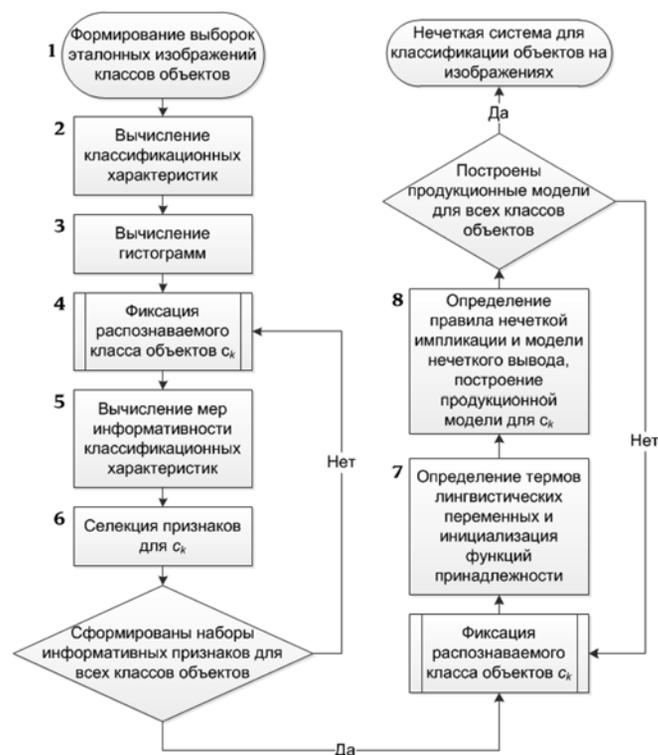


Рис. 1. Схема алгоритма для формирования нечеткой системы

5. Вычисляется разностная оценка прогноза разделимости τ^+ , характеризующая степень отличия между гистограммами всех одноименных ТП (j определяет вид ТП) для классов k и i :

$$\tau^+(k, i, j) = \sum_{l=1}^L \Delta_l^+(k, i, j),$$

где Δ_l^+ обозначает положительные значения интервальных оценок $\Delta_l(k, i, j) = \tilde{P}_l^k(j) - \tilde{P}_l^i(j)$.

При одинаковых значениях $\tau^+(k, i, j)$ и $\tau^+(k, i, q)$ различных ТП, $j \neq q$, вычисляются значения величин θ_{ki}^j и θ_{ki}^q , которые являются кардинальными числами индикаторных функций k -го класса, определенных на ТП j и q соответственно, и характеризуют "компактность" их представления, неоднородность и степень пересечения классов i и k по j -му ТП. Оценка θ_{ki}^j является числом длин серий неотрицательных значений интервальных оценок $\Delta_l(k, i, j)$ на области определения j -го текстурного признака или необходимым минимальным числом аналитических функций $f(t)$ для покрытия этих областей изменения ТП.

6. Формируются наборы информативных ТП по следующему критерию: ТП \tilde{T}_j считается информативным, если он имеет величину τ^+ больше, чем признаки \tilde{T}_q при $q \neq j$; в случае равных значений τ^+ наиболее информативным считается признак с меньшим значением $\theta_{k,i}^j$.

7. По результатам предыдущего этапа получены индивидуальные наборы информативных ТП для каждого класса. Теперь необходимо определить

значения лингвистических переменных (термы) для каждого признака из набора и соответствующие функции принадлежности (ФП), выражающие данные термы. В работе рассматриваются три способа инициализации ФП:

- классический способ задания функции на всей области определения значений ТП на основе средних значений по выборочным данным;
- интерполяция кусочно-линейной функцией

$$F_k^{(j)}(t) = \frac{t-t_0}{t_1-t_0} (\tilde{P}_k(t_1) - \tilde{P}_k(t_0)) + \tilde{P}_k(t_0),$$

где $F_k(t)$ — значение ФП k -го класса при значении текстурного признака t ; $\tilde{P}_k(t_1)$ и $\tilde{P}_k(t_0)$ — найденные значения относительных частот для значений текстурных признаков t_1 и t_0 соответственно;

с) кусочная аппроксимация набором функций $\{f_i(t)\}$. На рис. 2 показан пример такой аппроксимации для двух подтипов перистой облачности *Ci unc* (класс А) и *Ci floc* (класс В), для которых ФП имеют вид:

$$F_{A|B}^{(j)}(t) = \begin{cases} f_1(t; a = 0,45, c = 0,15), & 0,15 \leq t \leq 0,6, \\ f_2(t; a = 0,88, c = 0,07), & 0,8 \leq t \leq 0,95; \\ 0, & \text{при остальных значениях } t; \end{cases}$$

$$F_{B|A}^{(j)}(t) = \begin{cases} f_3(t; a = 0,1, c = 0,22), & 0,05 \leq t \leq 0,15, \\ f_4(t; a = 0,7, c = 0,085), & 0,6 \leq t \leq 0,8; \\ 0, & \text{при остальных значениях } t, \end{cases}$$

где $f_i(t; a_i, b_i)$ — гауссоиды, a и b — параметры. Границы задания функций $f_i(t)$ определяются на

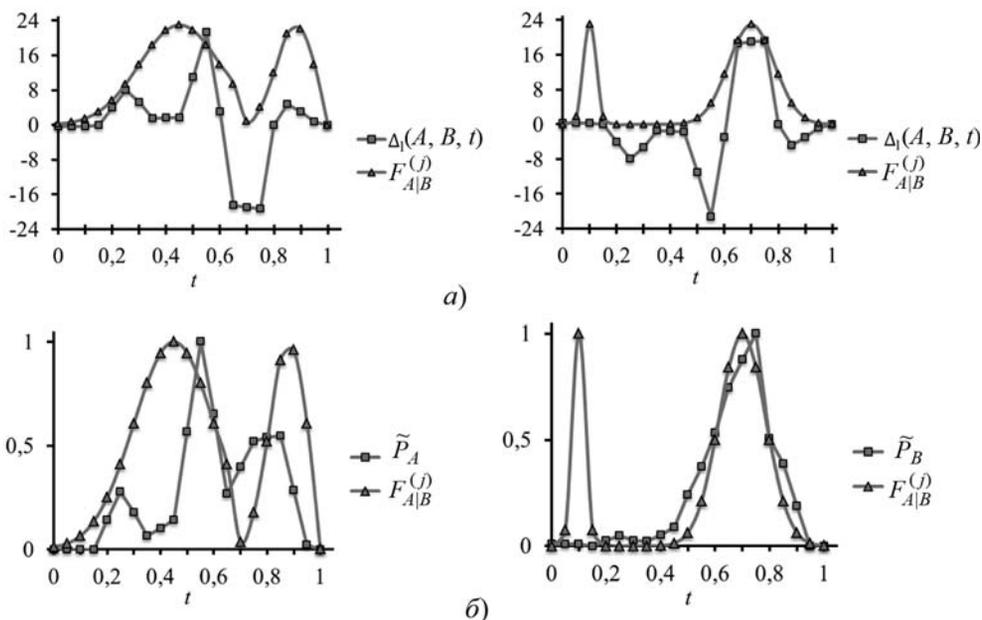


Рис. 2. Пример построения индикаторных функций на основе анализа разностных интервальных оценок (а) и аппроксимации нормальных ФП (б) для *Ci unc* (класс А) и *Ci floc* (класс В) гауссоидами

основе анализа значений интервальных оценок Δ_j^+ , $v = 1, \dots, \theta_{k,i}^j$. При этом ФП отражают преобладание класса $A(B)$ над классом $B(A)$, что обозначается индексом $A|B$ ($B|A$).

Дальнейшее уточнение границ областей определения кусочных ФП и их параметров происходит при адаптации нечеткой системы к обучающим данным. Задача интерпретации лингвистических термов может быть решена семантической декомпозицией кусочной ФП.

8. Продукционная модель представляет собой базу знаний и является конечной совокупностью правил нечетких продукций. Метод инициализации ФП кусочной аппроксимацией за счет покрытия базового терм-множества лингвистической переменной позволяет задавать простые правила нечеткого вывода:

$$\text{если } T_j^i \text{ — это } F_k^{(j)}(t_j), \text{ то } i = k. \quad (1)$$

Так как сформированы индивидуальные информативные наборы ТП, то наиболее подходящей моделью нечеткого вывода является Питтсбургский классификатор [12], правила которого имеют следующий вид.

Правило hi : если $\tilde{T}_1 = F_k^{(1)}$ и $\tilde{T}_2 = F_k^{(2)}$ и ...

$$\text{и } \tilde{T}_M = F_k^{(M)}, \text{ то класс} = c_i, w = Q_{hi}, \quad (2)$$

где hi — индекс правила h для i -го класса; Q_{hi} — вес правила или уровень доверия hi -му правилу, $Q_{hi} \in [0, 1]$; $j = \overline{1, M}$, M — число входных переменных (текстурных признаков); $i = \overline{1, K}$, K — число выходных переменных (распознаваемых классов); $F_k^{(i)}$ — термы с соответствующими функциями принадлежности, определенные на \tilde{T}_j .

В соответствии с формулами (1) и (2) термы по набору признаков, полученных для k -го класса, имеют уровень доверия $Q_{hk} = 1, \forall h, h = \overline{1, K}$, и связываются между собой логической операцией "И", формируя правило. Остальные термы по признакам, полученным для "фоновых" классов, также связываются операцией "И" и дополняют модель Питтсбургского классификатора для k -го класса, используя логическую операцию "ИЛИ". При этом правила имеют уровень доверия $Q_{hi} = 0,5, \forall i, i \neq k$.

В общем случае логические операции "И" и "ИЛИ" реализуют произвольные с учетом принципа двойственности Т-норму и Т-коформу соответственно [4, 13]. В модели Питтсбургского классификатора им соответствуют операции "умножения" и "суммирования" [12]. При этом точность нечеткого вывода достигается путем совместной работы

правил, относящихся к i -му классу, что выражено степенью принадлежности

$$R_i(\mathbf{x}^l) = \sum_{\text{Правило}_{hi}} \prod_{j=1}^M F_j^{(i)}(t_{jl}) Q_{hi} \quad (3)$$

где \mathbf{x}^l — вектор значений ТП для l -го обучающего образца.

Если при значениях ТП T_j^i , вычисленных по образцу \mathbf{x}^l , ФП $F_{k|i}^{(j)}(t_{jl}^k)$ возвращает значение $v = R_i(\mathbf{x}^l)$, то образец \mathbf{x}^l относится к классу k со степенью принадлежности, равной v , а выход классификатора определяется следующим образом:

$$y = \max_i R_i(\mathbf{x}^l). \quad (4)$$

Представленный алгоритм формирования нечеткой системы идентифицирует структуру и инициализирует параметры классификатора изображений. Для того чтобы повысить и оценить эффективность построенной системы, необходимо выполнить ее адаптацию к изменениям значений входных переменных с помощью генетической подстройки параметров ФП.

Алгоритм идентификации параметров нечеткой системы

Обучение нечеткой системы заключается в настройке значений параметров ФП на основе наборов эталонных изображений различных классов объектов таким образом, чтобы при предъявлении схожих по текстуре образцов нечеткая система относилась их с большей степенью принадлежности к определенному классу. Для этого используется генетический алгоритм обучения [13], при котором генотипом является набор хромосом для особи (конкретная реализация конфигурации нечеткой системы), представленный в виде совокупности ФП, входящих в состав нечеткой системы. Хромосомой является упорядоченная последовательность генов в виде набора адаптируемых параметров одной ФП. При этом аллель — это конкретное значение гена (параметра ФП). Оценка приспособленности хромосом осуществляется последовательной подстановкой аллелей каждой особи в ФП и вычислении в конце каждой эпохи фитнес-функции (ФФ) [14]:

$$FF_i(k) = \frac{1}{2} \sum_{l=1}^N (y(\mathbf{x}^l) - d^l)^2, \quad (5)$$

где k — номер эпохи; \mathbf{x}^l — вектор значений ТП для l -го обучающего образца; $y(\mathbf{x}^l) = R_i(\mathbf{x}^l)$ — степень принадлежности i -му классу (формула (3)); d^l —

эталонное значение степени принадлежности классу на этот образец, N — объем обучающей выборки $\left(N = \sum_{i=1}^K N_i \right)$. Критериями прекращения

(останова) обучения являются:

- реализация 100 эпох (по времени);
- достижение порогового значения $\Phi\Phi$, усредненной по классам,

$$FF(k) = \frac{1}{K} \sum_{i=1}^K FF_i(k) \leq 0,01,$$

где K — число классов;

- достаточно малое значение относительного изменения $FF_i(k)$ на соседних эпохах, равное 0,01.

Последний критерий в отличие от других является "частично завершающим" работу алгоритма, так как при его реализации фиксируются хромосомы, соответствующие i -му классу, и продолжается обучение для оставшихся классов.

Алгоритм генетического обучения для модели Питтсбургского классификатора, как следует из выражения (3), можно реализовать двумя способами:

1. Вычисление значений $\Phi\Phi$ согласно выражению (5) при $y(\mathbf{x}^l) = R_i(\mathbf{x}^l)$ с одновременной идентификацией всех параметров $\Phi\Phi$ (хромосом).

2. Идентификация параметров $\Phi\Phi$ отдельно для каждого правила нечеткого вывода. В этом случае генотипом является структура, включающая в себя только те $\Phi\Phi$, которые принадлежат одному правилу, а не вся конфигурация сети. В этом случае степень принадлежности $y(\mathbf{x}^l)$ определяется выражением

$$y(\mathbf{x}^l) = \prod_{l=1}^M F_{il}(t_{jl}^i) Q_{ij}.$$

При завершении работы алгоритма обучения принимаются хромосомы той особи, которая показывает наилучший результат классификации (наименьшее значение $\Phi\Phi$).

Вычислительные эксперименты и интерпретация их результатов

Набор эталонных изображений подтипов перистой облачности формировался путем применения методики сопоставления архивных данных наземных метеостанций со спутниковой съемкой MODIS [15]. При этом использовались данные 1-го спектрального канала (0,62...0,67 мкм) с пространственным разрешением 250 м за период 2005...2013 гг. в светлое время суток и при отсутствии снежного покрова для различных регионов планеты, полный список которых приведен в работе [16]. На основании информации о наличии перистой облачности из продукта MOD06_L2 осу-

ществлялось выделение эталонных образцов текстур на спутниковых снимках размером 21×21 пиксель. В результате были сформированы обучающая (400 фрагментов на каждый подтип перистой облачности) и тестовая (общее число 850 образцов) выборки.

Достоверность результатов классификации тестовых образцов определяется выражением

$$E_i = \frac{N_R^{(i)}}{N_T^{(i)}},$$

где $N_T^{(i)}$ — число тестовых фрагментов i -го класса

объектов; $N_R^{(i)}$ — число правильно проклассифицированных фрагментов тестовой выборки. При этом усредненная оценка вероятности правильной классификации

$$E_W = \sum_i \frac{N_R^{(i)}}{N_T^{(i)}},$$

где $N_T = \sum_i N_T^{(i)}$. При этом оценки $N_R^{(i)}$ рассчитаны с учетом выражения (4).

Для сравнения описанных выше способов реализации генетического алгоритма обучения нечеткой системы на основе Питтсбургского классификатора в качестве инварианта выбрано число изменений хромосом. При реализации только первого условия из вышеупомянутых критериев остановка затраченное время поиска значений оптимальных параметров $\Phi\Phi$ во второй схеме будет больше, чем в первой, а при реализации всех трех критериев остановка одновременно наблюдается обратная картина, что видно из представленных на рис. 3 графиков изменений усредненных значений $\Phi\Phi$ для каждой схемы.

Динамика изменений значений $\Phi\Phi$ при применении первой схемы обучения обусловлена тем, что при расчете $\Phi\Phi$ учитывается общий вклад правил, а не индивидуальные особенности каждого класса объектов, которые отражены в этих правилах. Поэтому при применении второй схемы обучения достигается большая эффективность классификации при увеличении числа рассматриваемых классов текстур, что подтверждается вычислительными экспериментами, результаты которых представлены в табл. 1. При этом кроме фрагментов изображений подтипов перистой облачности использовались эталонные изображения текстур кучевых, слоистых, кучево-дождевых, перисто-кучевых, высокослоистых и слоисто-дождевых облаков, а также ПП, описанные в работе [6]. Инициализация $\Phi\Phi$ осуществлялась кусочной аппроксимацией гауссовскими функциями.

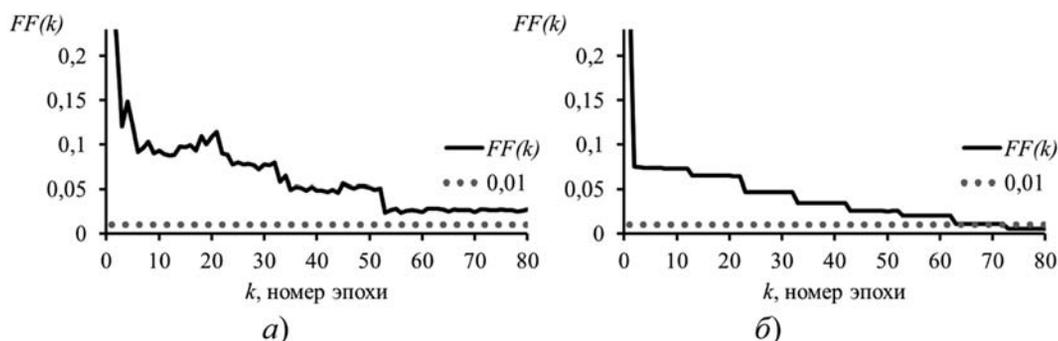


Рис. 3. Усредненные значения ФФ в процессе обучения нечеткой системы для классификации четырех подтипов перистой облачности по первой схеме (а) и по второй схеме (б)

Согласно предложенному алгоритму сформирован набор информативных ТП, который включает в себя шесть признаков, приведенных в табл. 2. При этом наибольшим значением оценки прогноза разделимости τ^+ и минимальным значением кардинального числа индикаторной функции $\theta_{k,i}^j$ для подтипа когтевидных перистых облаков (*Ci unc*) в сравне-

нии с другими подтипами позволяют сделать вывод об уникальности его текстуры.

На основе полученных наборов информативных ТП (табл. 2) и п. 8 предложенного алгоритма формирования нечетких систем создается база правил, представляющая собой цепочку рассуждений прямого вывода в виде продукционной модели, которая для когтевидных перистых облаков и произвольного образца k -го класса $x = \{T_1^k, T_2^k, T_3^k, T_4^k, T_5^k, T_6^k\}$ имеет следующий вид:

если

$$\left[\begin{array}{l} T_1^k \text{ это } F_{A|B}^{(1)} \text{ и } T_2^k \text{ это } F_{A|C}^{(2)} \text{ и } T_3^k \text{ это } F_{A|D}^{(3)}, Q=1,0 \\ \text{или} \\ T_1^k \text{ это } F_{A|B}^{(1)} \text{ и } T_4^k \text{ это } F_{A|B}^{(4)} \text{ и } T_5^k \text{ это } F_{A|B}^{(5)}, Q=0,5 \\ \text{или} \\ T_2^k \text{ это } F_{A|C}^{(2)} \text{ и } T_4^k \text{ это } F_{A|C}^{(4)} \text{ и } T_6^k \text{ это } F_{A|C}^{(6)}, Q=0,5 \\ \text{или} \\ T_3^k \text{ это } F_{A|D}^{(3)} \text{ и } T_5^k \text{ это } F_{A|D}^{(5)} \text{ и } T_6^k \text{ это } F_{A|D}^{(6)}, Q=0,5 \end{array} \right],$$

то $k = A$.

Для остальных подтипов перистой облачности продукционные модели формируются аналогично. Здесь и далее приняты следующие обозначения: А — когтевидные (*Ci unc*), В — хлопьевидные (*Ci floc*), С — послегрозовые (*Ci ing*), D — хребтовидные и перепутанные (*Ci int & Ci vert*) подтипы перистой облачности.

По результатам оценки достоверности результатов классификации при различных способах инициализации ФП для предложенной нечеткой системы классификации изображений подтипов перистой облачности была составлена табл. 3, в которой приведены значения E_i и вероятности ошибки классификации каждого класса. Наилучший результат достигается при кусочной аппроксимации функциями Гаусса, для которой оценка достоверности классификации $E_W = 0,58$. По результатам классификации тестовой выборки подтипов перистой об-

Таблица 1

Результаты классификации текстур различных типов облачности

Алгоритм обучения	E_W					
	Число классов (число информативных ТП)					
	2 (1)	3 (4)	4 (8)	6 (18)	8 (33)	10 (52)
1 схема	0,98	0,93	0,87	0,7	0,72	0,63
2 схема	0,98	0,93	0,92	0,77	0,85	0,81

Таблица 2

Наборы информативных ТП подтипов перистой облачности

Тип облачности	Текстурный признак			Информативность	
	Метод	Наименование	Обозначение	τ^+ , %	$\theta_{k,i}^j$
<i>Ci unc</i>	GLCM	Суммарное среднее (3, 90°)	T_1	55,25	6
	GLCM	Дисперсия (4, 45°)	T_2	62,25	5
	GLCM	Суммарное среднее (2, 90°)	T_3	58,75	7
<i>Ci floc</i>	GLCM	Суммарное среднее (3, 90°)	T_1	55,25	6
	GLCM	Момент обратной разности (4, 135°)	T_4	37	11
	SADH	Среднее S(2, 4)	T_5	35,25	15
<i>Ci ing</i>	GLCM	Дисперсия (4, 45°)	T_2	62,25	5
	GLCM	Момент обратной разности (4, 135°)	T_4	37	11
	SFR	Мода	T_6	37	18
<i>Ci vert & Ci int</i>	GLCM	Суммарное среднее (2, 90°)	T_3	58,75	7
	SADH	Среднее S(2, 4)	T_5	35,25	15
	SFR	Мода	T_6	37	18

лачности можно сделать вывод об их существенной неоднородности в наблюдаемых облачных полях на спутниковых снимках и переходе одних разновидностей облаков в другие.

Степень принадлежности классифицируемого фрагмента изображения к одному из классов определяется в соответствии с полученным значением выхода нечеткой системы по формуле (4). При этом сам вектор степеней принадлежности $R_i(x)$ образца к классам, значения которого получены по формуле (3), после его нормализации можно интерпретировать отдельно. В табл. 4 представлены результаты такой интерпретации для выборочных данных подтипов перистой облачности, где при условии, что значения степеней принадлежности R_i образца x отличаются не более чем на 0,1, он относится сразу к нескольким классам. Классификация происходит не только на исследуемые четыре подтипа перистой облачности, но и на их сочетания, например, класс АВ — когтевидные и/или хлопьевидные.

Таблица 3

Результаты классификации подтипов перистой облачности

Метод инициализации функции принадлежности	E_i				E_W
	Тип облачности				
	$Ci unc$	$Ci floe$	$Ci ing$	$Ci vert \& Ci int$	
Классический гауссовскими функциями	0,77	0,46	0,45	0,23	0,47
Классический треугольными функциями	0,74	0,48	0,26	0,36	0,47
Кусочная аппроксимация треугольными функциями	0,7	0,46	0,32	0,38	0,47
Линейная интерполяция	0,7	0,45	0,21	0,56	0,5
Кусочная аппроксимация гауссовскими функциями	0,72	0,52	0,5	0,55	0,58

Таблица 4

Результаты интерпретации выборочных данных подтипов перистой облачности

Маркер	Выборочные данные, %			
	$Ci unc$	$Ci floe$	$Ci ing$	$Ci int \& Ci vert$
N	7,47	9,34	7,52	6,85
A	66,55	13,29	7,97	11,05
AB	1,19	0,63	0,68	0,00
AC	0,17	0,16	0,46	0,00
AD	1,02	0,16	0,00	0,56
B	8,32	43,67	11,16	12,17
BC	0,34	1,27	2,05	0,84
BD	0,51	0,95	1,59	0,56
C	6,62	15,82	45,10	16,22
CD	0,51	0,79	1,59	1,82
D	7,30	13,92	21,87	49,93

При возникновении ошибки классификации или в случае, когда более двух значений степеней принадлежности R_i образца x близки по указанному правилу, фрагмент относится к классу N — не проклассифицировано.

Заключение

В работе предложен эффективный алгоритм классификации изображений на основе методов нечеткого моделирования, включающий в себя методику формирования наборов информативных ТП распознаваемых классов объектов и численный метод инициализации ФП лингвистических переменных путем кусочной аппроксимацией базового терм-множества значений ТП. На основе предложенного подхода впервые решена задача автоматической классификации изображений перистой облачности по четырем подтипам на основе текстурной информации с оценкой вероятности правильной классификации 0,58. Повышение эффективности классификатора при идентификации подтипов перистой облачности может быть достигнуто путем привлечения дополнительной информации об их физических параметрах, таких как высота и температура верхней границы облаков, оптическая толщина. Необходимо отметить, что при классификации полноразмерных снимков MODIS предложенной нечеткой системой используется маска перистой облачности, которую можно извлечь из продуктов MOD06_L2 и MYD06_L2.

Список литературы

1. **Вятчин Д. А.** Нечеткие методы автоматической классификации. Мн.: УП "Технопринт", 2004. 219 с.
2. **Код** для оперативной передачи данных приземных метеорологических наблюдений с сети станций РОСГИДРОМЕТА (КН-01 SYNOP) / Отв. ред. Н. П. Фахрутдинова. М.: Федеральная служба по гидрометеорологии и мониторингу окружающей среды (РОСГИДРОМЕТ), 2013. 79 с.
3. **Афонин С. В., Соломатов Д. В.** Методика учета оптико-метеорологического состояния атмосферы для решения задач атмосферной коррекции спутниковых ИК-измерений // Оптика атмосферы и океана. 2008. Т. 21, № 2. С. 147—153.
4. **Рутковская Д., Пилинский Д., Рутковский Л.** Нейронные сети, генетические алгоритмы и нечеткие системы: Пер с польск. И. Д. Рудинского. М.: Горячая линия — Телеком, 2006. 452 с.
5. **Гайдышев И.** Анализ и обработка данных: специальный справочник. СПб.: Питер, 2001. 725 с.
6. **Астафуров В. Г., Скороходов А. В.** Нейросетевой классификатор облачности по спутниковым данным // Информационные технологии. 2012. № 7. С. 32—37.
7. **Волкова Е. В., Успенский А. Б.** Оценки параметров облачного покрова по данным геостационарного МИСЗ МЕТЕОСАТ-9 круглосуточно в автоматическом режиме // Современные проблемы дистанционного зондирования Земли из космоса. 2010. Т 7. № 3. С. 65—73.
8. **Weszka J. S., Dyer C. R., Rosenfeld A. A.** Comparative Study of Texture Measures for Terrain Classification // IEEE Transaction on Systems, Man and Cybernetics. April 1976. Vol. SMC-6, N. 4. P. 269—285.
9. **Haralick R. M., Shanmugam K., Dinstein I.** Textural features for image classification // IEEE Transactions on Systems, Man and Cybernetics. November 1973. Vol. SMC-3, N. 6. P. 610—621.

10. Unser M. Sum and difference histograms for texture classification // IEEE Transaction on Systems, Pattern Analysis and Machine Intelligence. January 1986. Vol. PAMI-8, N. 1. P. 118–125.

11. Baum B. A., Tovinkere V., Titlow J., Welch R. M. Automated cloud classification of global AVHRR data using a fuzzy logic approach // Journal of Applied Meteorology. November 1997. Vol. 36. P. 1519–1540.

12. Smith S. F. A learning system based on genetic adaptive algorithms: Doctoral Dissertation. University of Pittsburgh, PA. 1980.

13. Осовский С. Нейронные сети для обработки информации / Пер. с польск. И. Д. Рудинского. М.: Финансы и статистика, 2002. 344 с.

14. Астафуров В. Г., Евсюткин Т. В. Программная система для автоматической классификации текстур подтипов перистой облачности на основе нейро-нечеткой сети // Докл. Томского государственного университета систем управления и радиоэлектроники. 2015. № 2 (36). С. 125–129.

15. Астафуров В. Г., Скороходов А. В. Сегментация спутниковых снимков облачности по текстурным признакам на основе нейросетевых технологий // Исследование Земли из космоса. 2011. № 6. С. 10–20.

16. Астафуров В. Г., Аксенов С. В., Евсюткин Т. В. Классификация перистой облачности по данным MODIS с помощью нечеткой нейронной сети // Современные проблемы дистанционного зондирования Земли из космоса. 2014. Т. 11, № 4. С. 265–275.

V. G. Astafurov^{1,2}, PhD, Full Professor, e-mail: astafurov@iao.ru,

T. V. Evsjutkin¹, PhD, Research Fellow, e-mail: oomphi.iao@gmail.com,

S. V. Axyonov^{2,3,4}, PhD, Assistant Professor, e-mail: axyonov@tpu.ru

¹V. E. Zuev Institute of Atmospheric Optics Russian Academy of Science, Siberian Branch, Tomsk, Russia

²Tomsk State University of Control Systems and Radio Electronics, Tomsk, Russia

³National Research Tomsk polytechnic university, Tomsk, Russia

⁴National Research Tomsk State University, Tomsk, Russia

An Effective Algorithm of Fuzzy System Design Based on Pittsburgh Classifier Model

The paper presents an effective algorithm based on fuzzy logic for low-contrast object classification in satellite images by the example of four subtypes of cirrus clouds: Cirrus uncinus (Ci unc), Cirrus floccus (Ci flocc), Cirrus incus-genitus (Ci ing), Cirrus vertebratus & Cirrus intortus (Ci vert & Ci int).

It demonstrates the description of proposed methods for fuzzy production rule base design and selection of informative texture feature sets. It considers some different methods of membership functions initialization, including a numerical method approximating the base term set of texture features. As a result of the research we proposed the automated low-contrast texture object classifier based on fuzzy inference model known as the Pittsburgh classifier. The effectiveness of the proposed algorithm is demonstrated for the image classification task for four cirrus cloud subtypes using MODIS satellite data.

The paper describes a modification of genetic algorithm for Pittsburgh classifier's parameters adaptation. The computer experiments used MODIS data containing texture images of different cloud types and demonstrated the proposed modification gets better results as compared to the classical approach.

The paper presents the suggested set of informative features of the mentioned cirrus cloud subtypes, the developed production model of the Ci unc cloud subtype and the estimation of probability of correct classification that reaches 0.58 by implementing of membership function initialization by piecewise approximation with Gauss functions. We interpreted the classification outcomes of the test cirrus cloud subtype sets and concluded the proposed fuzzy model is able to identify the translucent cover of cirrus cloud fields.

Keywords: classification, fuzzy systems, texture, Pittsburgh classifier, cirrus cloud, genetic training

References

1. Vjatchenin D. A. *Nechetkie metody avtomaticheskoy klassifikacii* (Fuzzy methods of automated classification). Minsk.: UP "Tehnoprint", 2004. 219 p. (in Russian).

2. *Kod dlja operativnoj peredachi dannyh prizemnyh meteorologicheskikh nabljudenij s seti stancij ROSGIDROMETA (KN-01 SYNOP)* (A code of operate meteorological data transmission of ROSGIDROMET network (KN-01 SYNOP)). Otv. red. N. P. Fahrutdinova. Moscow, Federal'naja sluzhba po gidrometeorologii i monitoringu okruzhajushhej sredy (ROSGIDROMET), 2013. 79 p. (in Russian).

3. Afonin S. V., Solomatov D. V. Metodika ucheta optiko-meteorologicheskogo sostojanija atmosfery dlja reshenija zadach atmosfernoj korekcii sputnikovyh IK-izmerenij (A method of atmosphere optical and meteorological status analysis for satellite infrared spectroscopy correction tasks, *Optika atmosfery i okeana*, 2008, vol. 21, no. 2, pp. 147–153 (in Russian).

4. Rutkovskaja D., Pilin'skij D., Rutkovskij L. *Nejronnye seti, geneticheskie algoritmy i nechetkie sistemy* (Neural networks, genetic algorithms and fuzzy systems). Per s pol'sk. I. D. Rudinskogo, Moscow, Gorjachaja linija — Telekom, 2006, 452 p. (in Russian).

5. Gajdshev I. *Analiz i obrabotka dannyh: special'nyj spravocnik*. (Analysis and data processing: special reference), SPb: Piter, 2001, 725 pp. (in Russian).

6. Astafurov V. G., Skorohodov A. V. Nejrosetevoj klassifikator oblachnosti po sputnikovym dannyh (Neural classifier of clouds based on remote sensing data), *Informacionnye tehnologii*, 2012, no. 7, pp. 32–37 (in Russian).

7. Volkova E. V., Uspenskij A. B. Ocenki parametrov oblachnogo pokrova po dannyh geostacionarnogo MISZ METEOSAT-9 kruglosutochno v avtomaticheskom rezhime (Cloud cover parameters' assessment using data of METEOSAT-9, the small satellite of remote sensing), *Sovremennye problemy distancionnogo zondirovanija Zemli iz kosmosa*, 2010, vol. 7, no. 3, pp. 65–73 (in Russian).

8. Weszka J. S., Dyer C. R., Rosenfeld A. A. Comparative Study of Texture Measures for Terrain Classification, *IEEE Transaction on Systems, Man and Cybernetics*. April 1976. vol. SMC-6, no. 4. pp. 269–285.

9. Haralick R. M., Shanmugam K., Dinstein I. Textural features for image classification, *IEEE Transactions on Systems, Man and Cybernetics*, November 1973, vol. SMC-3, no. 6, pp. 610–621.

10. **Unser M.** Sum and difference histograms for texture classification, *IEEE Transaction on Systems, Pattern Analysis and Machine Intelligence*, January 1986, vol. PAMI-8, no. 1, pp. 118–125.

11. **Baum B. A., Tovinkere V., Titlow J., Welch R. M.** Automated cloud classification of global AVHRR data using a fuzzy logic approach, *Journal of Applied Meteorology*, November 1997, vol. 36, pp. 1519–1540.

12. **Smith S. F.** *A learning system based on genetic adaptive algorithms*: Doctoral Dissertation. University of Pittsburgh, PA. 1980.

13. **Osovskij S.** *Нейронные сети для обработки информации* (Neural networks for information processing). Per. s pol'sk. I. D. Rudinskogo, Moscow: Finansy i statistika, 2002. 344 p. (in Russian).

14. **Astafurov V. G., Evsjutkin T. V.** Программная система для автоматической классификации текстур подтипов перистой облачности на

основе нейро-нечеткой сети (Software for automated cirrus cloud subtypes texture classification based on neuro-fuzzy network), *Doklady Tomского государственного университета систем управления и радиоэлектроники*, 2015, no. 2 (36), pp. 125–129 (in Russian).

15. **Astafurov V. G., Skorohodov A. V.** Segmentacija sputnikovyh snimkov oblachnosti po teksturnym priznakam na osnove nejrosetevyh tehnologij (Satellite image segmentation using texture features based on neural technologies), *Issledovanie Zemli iz kosmosa*. 2011. no. 6, pp. 10–20 (in Russian).

16. **Astafurov V. G., Axyonov S. V., Evsjutkin T. V.** Klassifikacija peristoj oblachnosti po dannym MODIS s pomoshh'ju nechetkoj nejronnoj seti (Cirrus cloud classification using MODIS data with help of neuro-fuzzy network), *Sovremennye problemy distancionnogo zondirovaniya Zemli iz kosmosa*, 2014, vol. 11, no. 4, pp. 265–275 (in Russian).

УДК 629.05

М. Ф. Волобуев, канд. техн. наук, докторант, e-mail: volmf81@mail.ru,
А. М. Мальцев, канд. техн. наук, доц., ст. науч. сотр.,
И. В. Соколов, преподаватель, e-mail: ig-tan@yandex.ru,
Военный учебно-научный центр ВВС "Военно-воздушная академия
имени проф. Н. Е. Жуковского и Ю. А. Гагарина", г. Воронеж

Экспериментальные исследования макета резервированной системы, управляемой нечетким контроллером

Рассмотрены вопросы практической реализации на современной элементной базе устройств обнаружения отказов в резервированных системах с использованием нечеткой логики, а также минимального числа термов входной лингвистической переменной, позволяющих обнаруживать отказы в резервированных системах.

Ключевые слова: макет, нечеткая логика, резервированная система, терм

Введение

Перспективным направлением повышения надежности технических систем различного назначения является резервирование наиболее ответственных каналов (блоков, устройств) с нечеткой логикой обнаружения отказов и управления резервированной системой, в соответствии с которой осуществляются выбор и подключение одного из каналов системы к ее выходу. Теоретические аспекты создания таких резервированных систем и алгоритмы функционирования нечетких логических управляющих устройств представлены в статьях [1–3].

Целесообразность применения методов нечеткой логики в резервированных системах подтверждена результатами статистического имитационного моделирования процесса функционирования резервированной системы с нечетким контроллером [2–6]. Было показано, что при функционировании в условиях полной определенности нечеткий контроллер не уступает пороговому устройству по вероятности правильного обнаружения отказов. Если же функционирование резервированной системы происходит в условиях неопределенности, то проявля-

ется свойство робастности нечеткого контроллера, которое обеспечивает его превосходство перед пороговым устройством.

Однако вопросы практической реализации устройств обнаружения отказов, использующих нечетко-множественный подход, на основе возможностей современной микроэлектронной техники оставались открытыми. Также остался открытым вопрос о минимальном числе термов базового терм-множества входной лингвистической переменной, позволяющих обнаруживать отказы в резервированных системах.

Описание экспериментального макета

Рассмотрим возможность технической реализации способа обнаружения отказов с использованием нечеткой логики. Для этого был разработан макет двухканальной резервированной системы, с помощью которого проводились экспериментальные исследования. Для проверки возможности реализации нечеткой логики на широко распространенной элементной базе был выбран самый простой 8-битный микроконтроллер PIC16F628, а програм-

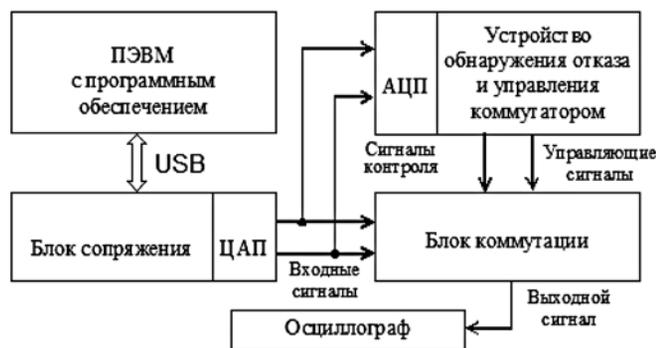


Рис. 2. Схема экспериментального стенда

ма была осуществлена с использованием языка программирования Ассемблер, который позволяет считать число операций в микроконтроллере.

Внешний вид макета показан на рис. 1 (см. третью сторону обложки), структурная схема — на рис. 2.

На структурной схеме показаны в виде условных изображений и обозначений составные части изделия и связи между ними.

Макет состоит из математической и натурной части.

Математическая часть: математическая модель формирования цифровых сигналов; драйверы для связи математической модели с устройством преобразования цифровых сигналов в аналоговые, нечеткая модель обнаружения отказа в канале и формирования управляющих сигналов для коммутации каналов.

Натурная часть: аналого-цифровой преобразователь (АЦП) ADC7998; микроконтроллер PIC16F628A для реализации нечетких алгоритмов обнаружения отказа и формирования управляющих сигналов; оптореле КР239КП3А; блок питания 5 В; жидкокристаллический дисплей LCD Winstar 20×4; устройство памяти EEPROMAT1024С.

Входные сигналы формировались на ПЭВМ в программной среде *MATLAB/Simulink* в цифровом виде с последующим их преобразованием в аналоговые с помощью цифроаналогового преобразователя (ЦАП), построенного на программируемой электронной плате *ArduinoMega2560*. Данная модель позволяет генерировать на выходе требуемые аналоговые сигналы и изменять их значения в процессе работы в пределах 0...5 В.

На базе микроконтроллера была реализована нечеткая модель обнаружения отказа в канале и формирования управляющих сигналов в соответствии с упрощенным алгоритмом нечеткого вывода [7].

Алгоритм обнаружения отказов с использованием нечеткой логики следующий: измерение входных сигналов; сравнение их с использованием нечеткой логики; выбор доминирующих альтернатив и подключение одного из каналов в соответствии с доминирующими альтернативами.

В качестве алгоритма нечеткого вывода используется наиболее часто встречающийся на практике алгоритм Мамдани.

Данный алгоритм является одним из первых, который нашел применение в системах нечеткого вывода. Формально алгоритм Мамдани может быть представлен в виде рис. 3 [7].

Входными сигналами для контроллера на основе нечетко-множественного подхода являются выходные сигналы каналов 1 и 2. Считалось, что i -й канал системы исправен, если значение его выходного параметра x находится в рабочем диапазоне $\delta = (m_x \pm 3\sigma_x)$, где σ_x — среднее квадратическое отклонение выходного значения параметра от математического ожидания m_x , и отказавшим, если значение вышло за пределы рабочего диапазона в ту или иную сторону.

С учетом использования в макете нечеткого контроллера задается следующий диапазон допустимых значений, относительно которых строятся функции принадлежности (ФП) термов лингвистической переменной: $\beta = (m_x - 2\sigma_x)$; $c = (m_x + 2\sigma_x)$; $\alpha = (m_x - 4\sigma_x)$; $\gamma = (m_x + 4\sigma_x)$.

В качестве функции принадлежности терм-множеств входных лингвистических переменных используются трапециевидные функции *LR* и *LR1* рис. 4 [7], где $LR = (\beta, c, \alpha, \gamma)$ и $LR1 = (\beta, c, \alpha, \gamma)$. Для *LR*-функции β, c — это точки, в которой значение функции принадлежности равно единице; $\alpha(\gamma)$ — координаты ближайшей слева (справа) к $x = \beta$ ($x = c$) точки базовой шкалы, в которой функция принадлежности принимает нулевое значение. Трапециевидная функция *LR1* представляет

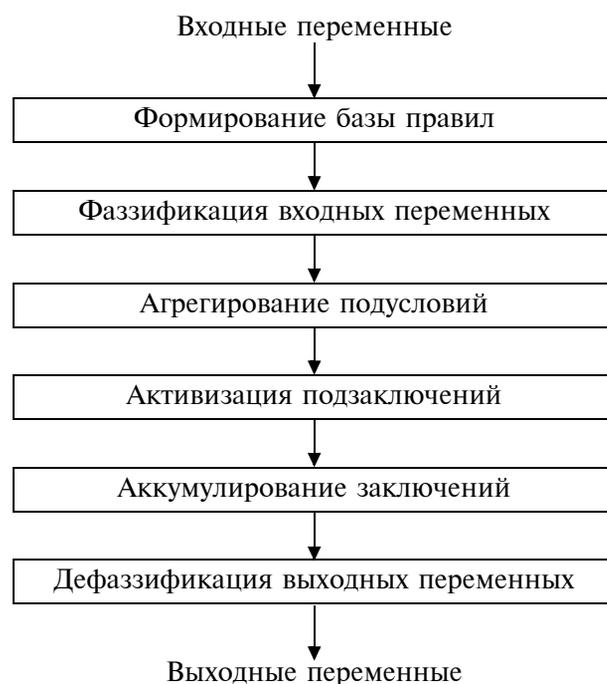


Рис. 3. Диаграмма процесса Мамдани

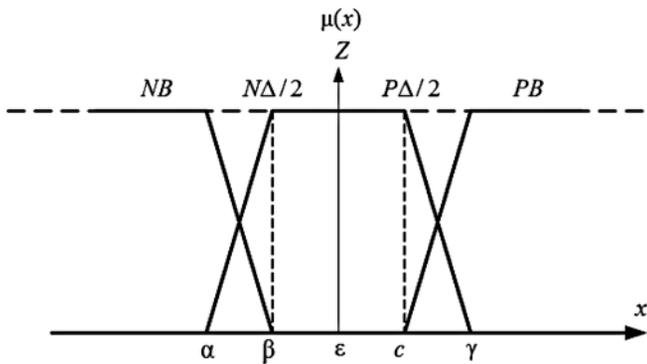


Рис. 4. Вид трапецевидных LR- и LR1-функций принадлежности качественных оценок отклонений выходных значений параметров

собой перевернутое изображение LR-функции. Поэтому для LR1 функции α и γ — точки, в которых функция принадлежности равна единице, а интервал $[\beta, c]$ соответствует нулевым значениям функции принадлежности.

Вид термов входной лингвистической переменной представлен на рис. 4 [7], где первая буква в обозначении указывает знак (сторону отклонения) числовой переменной (N — отрицательное, P — положительное), вторая буква говорит об абсолютном значении переменной (B (*big*) — большое, S (*small*) — маленькое), а символ Z соответствует попаданию переменной в предел от β до c , при котором однозначно считается, что элемент работает безотказно [7].

Для того чтобы различать такие понятия в степени уверенности решений, как "малая уверенность" (S), "достаточная уверенность" (M) и "сильная уверенность" (B), представим для каждого из четырех принимаемых решений перечисленные понятия в виде трех термов выходной лингвистической переменной, показанных на рис. 5.

Под нечетким управлением будем понимать процедуры принятия решений, представленные в виде некоторой совокупности правил, основанных на логико-формальном представлении знаний о функционировании системы (в лингвистической форме) и преобразовании их с применением методов нечеткой логики. Эти правила описывают принятие решений по управлению и формируются квалифицированными экспертами в области логического резервирования систем.

В соответствии с диаграммой, представленной на рис. 3, значения выходных цифровых сигналов, формируемых с помощью математической модели, поступают в систему нечеткого вывода.

В системах нечеткого вывода используется база правил нечетких продукций, в которых условия и заключения сформулированы в терминах нечетких лингвистических высказываний. База правил нечетких продукций представляет собой конечное множество правил нечетких продукций, согласованных

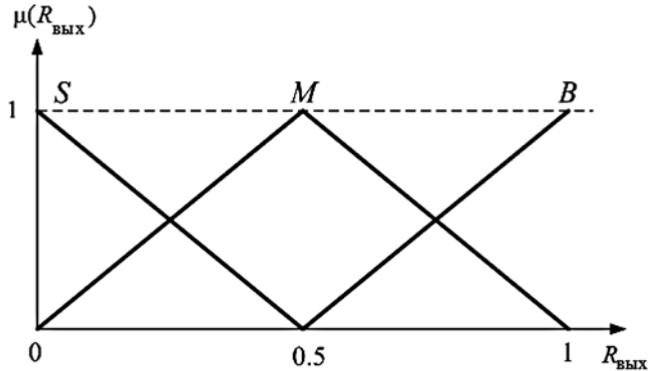


Рис. 5. Вид функций принадлежности качественных оценок уверенности в выбранных решениях по управлению системой

относительно используемых в них лингвистических переменных.

Наиболее часто база правил представляется в форме структурированного текста [7]:

ПРАВИЛО 1:

ЕСЛИ "Условие-1" ТО "Заключение-1(F1).

В качестве условий и заключений правил могут использоваться только нечеткие лингвистические высказывания, причем в каждом из нечетких высказываний должны быть определены функции принадлежности значений терм-множества для каждой из лингвистических переменных.

Для спроектированной двухканальной системы сформируем множество состояний (условий) и решений, которые должны приниматься в процессе функционирования системы. К такому множеству будут относиться следующие состояния (условия):

- 1) выходные параметры каналов 1 и 2 находятся в допуске;
- 2) выходной параметр канала 1 в допуске, канала 2 — положительно большой;
- 3) выходной параметр канала 1 в допуске, канала 2 — отрицательно большой;
- 4) выходные параметры каналов 1 и 2 положительно большие;
- 5) выходной параметр канала 1 положительно большой, канала 2 — отрицательно большой;
- 6) выходные параметры каналов 1 и 2 отрицательно большие;
- 7) выходной параметр канала 1 отрицательно большой, канала 2 — положительно большой;
- 8) выходной параметр канала 1 отрицательно большой, канала 2 — в допуске;
- 9) выходной параметр канала 1 положительно большой, канала 2 — в допуске.

В качестве решения (заключения) используются четыре правила, составляющие полную группу событий: I — канал 1 и канал 2 исправны; II — канал 1 неисправен, канал 2 исправен; III — канал 1 исправен, канал 2 неисправен; IV — оба канала 1 и 2 неисправны.

Например, подобным правилом, соответствующим принимаемому решению I и записанным на естественном языке для рассматриваемой двухканальной системы, может быть следующее высказывание: "Если отклонения фактического значения выходного параметра канала 1 и канала 2 соответствуют попаданию переменной в допустимый предел (рабочий диапазон значений) выходного параметра от математического ожидания (m_x), при котором считается, что элемент работает исправно — Z , то система исправна".

С использованием логико-формальных методов и введенных выше терм-множеств входных и выходных лингвистических переменных это правило в соответствии с упрощенным алгоритмом нечеткого вывода [7] записывается в следующем виде:

$$\begin{aligned} & ((P_{k1} = Z) \wedge (P_{k2} = Z)) \mapsto \\ & \mapsto ((I = B) \wedge (II = B) \wedge (III = S) \wedge (IV = S)). \end{aligned}$$

Аналогичным образом записываются соответствующие логико-формальные правила (правила нечеткого вывода) для принятия решений 1...9:

$$\begin{aligned} & ((P_{k1} = Z) \wedge (P_{k2} = Z)) \mapsto \\ & \mapsto ((I = B) \wedge (II = B) \wedge (III = S) \wedge (IV = S)); \\ & ((P_{k1} = Z) \wedge (P_{k2} = PB)) \mapsto \\ & \mapsto ((I = S) \wedge (II = S) \wedge (III = B) \wedge (IV = S)); \\ & ((P_{k1} = Z) \wedge (P_{k2} = NB)) \mapsto \\ & \mapsto ((I = S) \wedge (II = S) \wedge (III = B) \wedge (IV = S)); \\ & ((P_{k1} = PB) \wedge (P_{k2} = PB)) \mapsto \\ & \mapsto ((I = S) \wedge (II = S) \wedge (III = S) \wedge (IV = B)); \\ & ((P_{k1} = PB) \wedge (P_{k2} = NB)) \mapsto \\ & \mapsto ((I = S) \wedge (II = S) \wedge (III = S) \wedge (IV = B)); \\ & ((P_{k1} = NB) \wedge (P_{k2} = NB)) \mapsto \\ & \mapsto ((I = S) \wedge (II = S) \wedge (III = S) \wedge (IV = B)); \\ & ((P_{k1} = NB) \wedge (P_{k2} = PB)) \mapsto \\ & \mapsto ((I = S) \wedge (II = S) \wedge (III = S) \wedge (IV = B)); \\ & ((P_{k1} = NB) \wedge (P_{k2} = Z)) \mapsto \\ & \mapsto ((I = S) \wedge (II = B) \wedge (III = S) \wedge (IV = S)); \\ & ((P_{k1} = PB) \wedge (P_{k2} = Z)) \mapsto \\ & \mapsto ((I = S) \wedge (II = B) \wedge (III = S) \wedge (IV = S)), \end{aligned} \quad (1)$$

где \wedge — знак логической операции "И"; \mapsto — знак логической операции "секвенция", обозначающий, что высказывание, стоящее в правой части выражения, "истинно" тогда и только тогда, когда "истинно" высказывание в его левой части; $P_{k1, 2}$ — значения формируемого выходного сигнала.

Для нечетких подмножеств (1) степень выполнения условия "Если" будет определять степень применения данного правила. Поэтому степень уверенности в выборе того или иного решения будет определяться весом, соответствующим весу функции принадлежности пересечения нечетких подмножеств NB, Z, PB в условии "Если".

Следующим шагом в алгоритме Мамдани является операция фаззификации — процедура нахождения значений функций принадлежности нечетких множеств (термов) на основе обычных (не нечетких) исходных данных. Этот этап часто называют приведением к нечеткости.

Вычисление значений ФП терма, представленной в виде нечеткого числа, заданного упорядоченной четверкой LR (см. рис. 4), выполняется по формуле [7]

$$\mu(x) = \begin{cases} 0, & x \leq \alpha, \\ (x - \alpha)/(\beta - \alpha), & \alpha \leq x \leq \beta; \\ 1, & \beta \leq x \leq c, \\ (\gamma - x)/(\gamma - c), & c \leq x \leq \gamma; \\ 0, & x \geq \gamma, \end{cases} \quad (2)$$

а вычисление значений ФП-терма в виде нечеткого числа, заданного упорядоченной четверкой LRI (см. рис. 4) по формуле [7]:

$$\mu_1(x) = \begin{cases} 1, & x \leq \alpha, \\ (\beta - x)/(\beta - \alpha), & \alpha \leq x \leq \beta; \\ 0, & \beta \leq x \leq c, \\ (x - c)/(\gamma - c), & c \leq x \leq \gamma; \\ 1, & x \geq \gamma. \end{cases} \quad (3)$$

Поскольку в модели во всех правилах в качестве логической связки для подусловий применяется только нечеткая конъюнкция (операция "И"), то в качестве метода агрегирования используется операция \min -конъюнкции [7].

$$\mu_{P_{k1} \cap P_{k2}} = \min(\mu_{P_{k1}}(x), \mu_{P_{k2}}(x)). \quad (4)$$

Тем самым операция агрегирования представляет собой процедуру определения степени истинности условий по каждому из правил системы нечеткого вывода.

Этап активизации (активации) в используемом алгоритме Мамдани представляет собой процедуру нахождения степени истинности каждого из подзаключений правил нечетких продукций. В результате выполнения данной операции получаем сумму весовых коэффициентов для различных решений.

Выражение (4) представляет собой матрицу β , получаемую при суммировании весовых коэффициентов для различных решений:

$$\beta = \begin{bmatrix} \beta_{SI} & \beta_{SII} & \beta_{SIII} & \beta_{SIV} \\ \beta_{MI} & \beta_{MII} & \beta_{MIII} & \beta_{MIV} \\ \beta_{VI} & \beta_{VII} & \beta_{VIII} & \beta_{VIV} \end{bmatrix}, \quad (5)$$

где $\beta_{S, M, B(i)}$ — сумма весовых коэффициентов выходных лингвистических переменных (решений) в выражении (5).

Операция дефаззификации — процесс нахождения обычного (не нечеткого) значения для каждой из выходных лингвистических переменных. Рассчи-

тывается по методу центра тяжести по формуле [7]

$$F_j = \frac{\sum_{i=1}^n V_i y_i}{\sum_{i=1}^n V_i}, \quad (6)$$

где n — число правил вывода; y_i — коэффициент (1 — для значения B , 0 — для значения S), j — номер вывода.

Решением является лингвистическая переменная с максимальным значением степени истинности F_j .

Алгоритм реализации модели работы контроллера на основе нечетко-множественного подхода представлен на рис. 6.

Следует также отметить, что в нечетком контроллере использовался упрощенный алгоритм Мамдани, без выполнения операции аккумуляции заключений и выполнения операции дефаззификации с помощью метода определения центра тяжести для одноточечных множеств (определение среднего) вместо метода центроида площади, где для вычисления необходимо выполнять интегральные вычисления.

На схеме рис. 6 в блоках алгоритма реализации работы контроллера на основе нечетко-множественного подхода показано:

- блоки (3—7) — границы интервала нахождения сигнала с выходов 1-го и 2-го каналов;
- блоки (8—12) — с использованием формул (2) и (3) вычисление значений ФП при выполнении операции фаззификации;
- блок 13 — выполнение операции агрегирования с использованием формулы (4);
- блок 14 — выполнение операции активизации с учетом использования выражения (5);
- блок 15 — выполнение операции дефаззификации с использованием выражения (6);
- блок 16 — выбор решения о состоянии системы по максимальному критерию.

Результаты экспериментальных исследований

С использованием макета были проведены исследования для определения возможности нечеткого контроллера по обнаружению мгновенных отказов, возникающих, например, при коротком замыкании и обрыве цепи. На рис. 7 представлены зависимости вероятности безотказной работы сис-

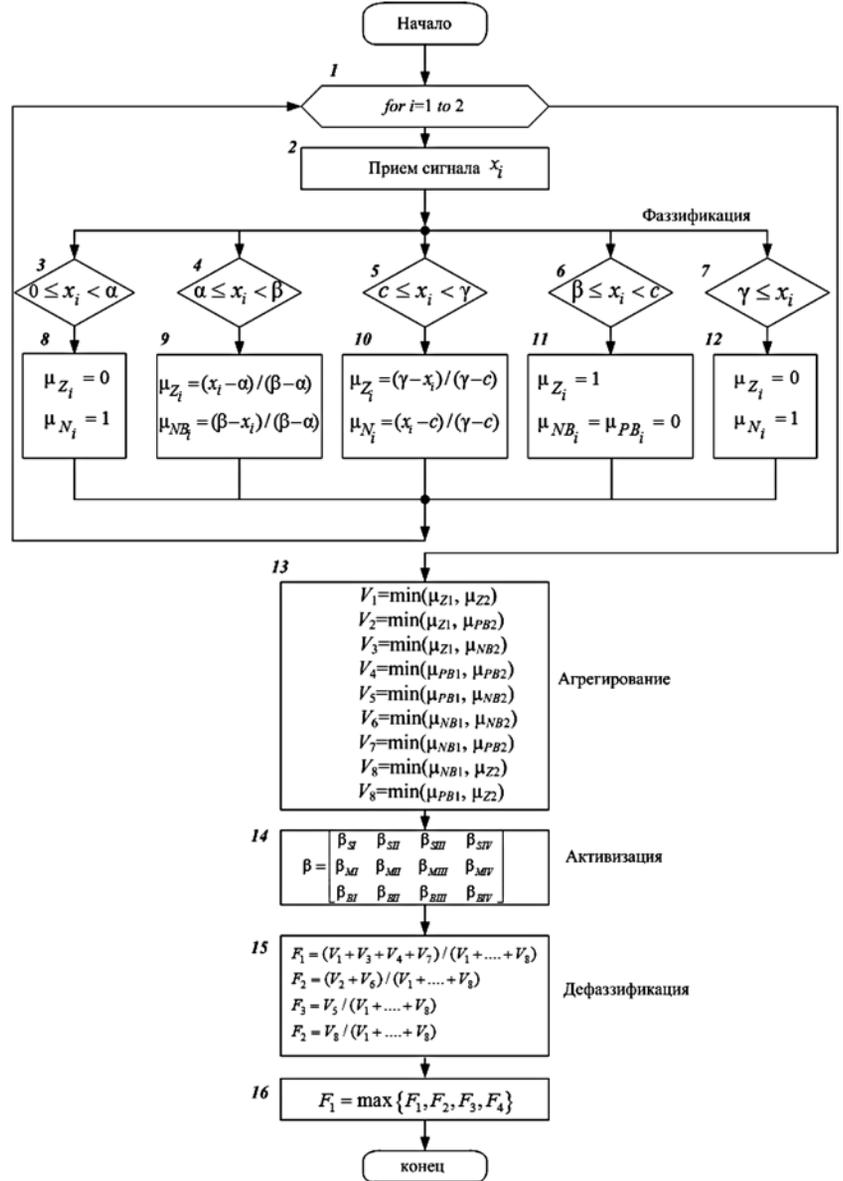


Рис. 6. Алгоритм реализации модели работы контроллера на основе нечетко-множественного подхода

темы $P_{\text{БРС}}$ от вероятности безотказной работы каждого канала $P_{\text{БРС}1}$ для идеальной двухканальной резервированной системой с замещением (сплошная линия), реализованной в программе MATLAB Simulink, и системы, реализованной в макете, управляемой нечетким контроллером (точки).

Как видно из рис. 7, системы сопоставимы, небольшие отличия в вероятностях безотказной работы систем обусловлены дискретностью преобразования сигналов платой ArduinoUno. Можно сделать вывод, что система, управляемая нечетким контроллером, способна обнаруживать мгновенные отказы и правильно управлять переключением каналов.

По проведенным исследованиям полный цикл программы 8-битного контроллера составил 10 800

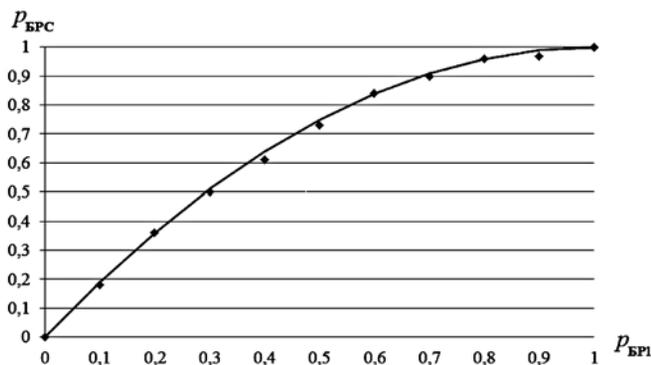


Рис. 7. Зависимость вероятности безотказной работы резервированной системы $P_{БРС}$ от вероятности безотказной работы каждого канала $P_{БР1}$

операций при частоте 4 МГц. Длительность одной операции составила ~ 1 мкс, общее время на выполнение всех операций $\sim 0,1$ с. Увеличение числа термов незначительно сказывается на числе выполняемых операций, но значительно увеличивает объем занимаемой программной памяти, так как добавляется по 2700 байт на каждый новый отрезок функции принадлежности.

Экспериментальные исследования с применением термов в виде экспоненциальных функций не проводились, так как это привело бы к усложнению вычисления значений термов. Например, при разложении экспоненты в ряд Тейлора вычисление каждого члена ряда требует 2800 операций при частоте микроконтроллера 4 МГц. При этом увеличение точности вычисления экспоненты на один член ряда сопоставимо по количеству занимаемой памяти с добавлением одного отрезка в кусочно-линейной аппроксимации для одного из термов, а число операций, выполняемых при каждом цикле алгоритма, увеличивается на 2800 при увеличении времени на 3 мс.

Для реализации более сложных программ и алгоритмов число операций также увеличивается, поэтому целесообразно применять 16- и 32-битные микроконтроллеры. В состав 16-битного контроллера входит 16-битное арифметико-логическое устройство (АЛУ), которое позволяет выполнять за один командный такт следующие операции: сложение, вычитание, битовый сдвиг и поразрядные логические операции, включая инверсию. Данное АЛУ выполняет операции с 16-битными словами или байтами в зависимости от синтаксиса инструкции. В состав ядра 16-битного АЛУ входит умножитель 17×17 бит, который позволяет выполнять операции умножения 16×16 , 16×8 и 8×8 бит, знаковые и беззнаковые за один командный такт. А в рассматриваемом 8-битном контроллере используется однобитное АЛУ, которое может выполнять операцию последовательного сдвига при выполнении операций умножения и сложения. В связи

с этим при переходе к 16-битному контроллеру число операций, необходимых для деления и умножения, сокращается до 19 вместо 1200 при использовании 8-битного контроллера, соответственно, а весь алгоритм, рассмотренный выше, выполняется за 300 операций. Переход к 32-битному контроллеру не влияет на скорость выполнения алгоритма кусочно-линейной аппроксимации, но позволяет находить члены ряда экспоненты вплоть до 8 (16-битный лишь до 4) за одну операцию.

В настоящее время имеются микропроцессоры, которые аппаратно поддерживают нечеткие модели [8]. Например, процессор *CPU12* аппаратно поддерживает операции "нечеткой логики", что позволяет эффективно реализовывать на базе микроконтроллеров семейства *HC12* соответствующие системы контроля. Группа команд "нечеткой логики" включает четыре команды. Команда *MEM* выполняет фаззификацию, преобразуя точные значения входных переменных в значения лингвистических переменных в соответствии с функциями принадлежности из базы знаний. Далее с помощью команд *REVV* и *REV* проводится непосредственно обработка полученных значений по заданному алгоритму с учетом или без учета весовых коэффициентов правил обработки. С помощью команды *WAV* осуществляется дефаззификация, в результате которой выполняется переход от нечетких значений выходной переменной к точным выходным значениям, обеспечивающим требуемое управление системой. Команды имеют длину от 1 до 6 байт. Выполнение большинства команд занимает от 2 до 6 тактов. Больше время выполнения имеют команды деления (11...12 тактов), умножения с накоплением (13 тактов) и прерывания (8...11 тактов).

Как видно из приведенных примеров, неспециализированные контроллеры по времени выполнения алгоритма, изложенного выше, сопоставимы со специализированными "нечеткими" контроллерами.

Оценка минимального числа терм-множеств

Минимальное число термов входной лингвистической переменной оценивали с использованием машинного эксперимента в программе *MATLAB Simulink*, в которой реализован широко известный алгоритм Мамдани [7], с аналогичными рассмотренными выше правилами нечетких продукций, которые для краткости не приводятся. То есть считалось, что i -й канал системы исправен, если значение его выходного параметра x находится в рабочем диапазоне, и отказавшим, если значение вышло за пределы рабочего диапазона в ту или иную сторону.

При этом рассматривались терм-множества входной лингвистической переменной с 3, 5 и 7 термами, вид которых представлен на рис. 4, 8, а и 8, б, соответственно, и выходной с 3 термами (рис. 9). Для входной лингвистической переменной из 3 и 5 термов было принято: $\alpha = (m_x - 6\sigma_x)$, $\beta = (m_x - 3\sigma_x)$,

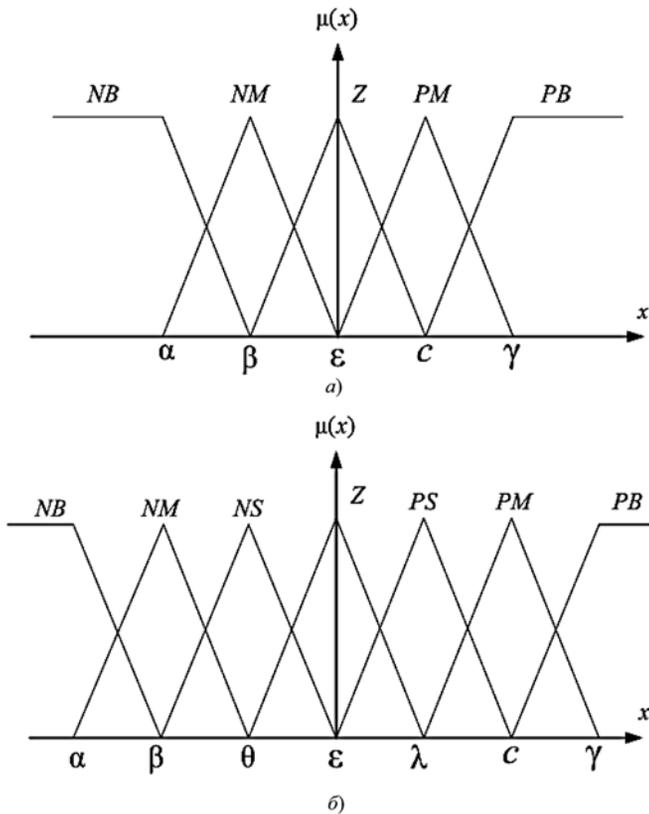


Рис. 8. Вид функций принадлежности терм-множеств входной лингвистической переменной с 5 термами (а) и с 7 термами (б)

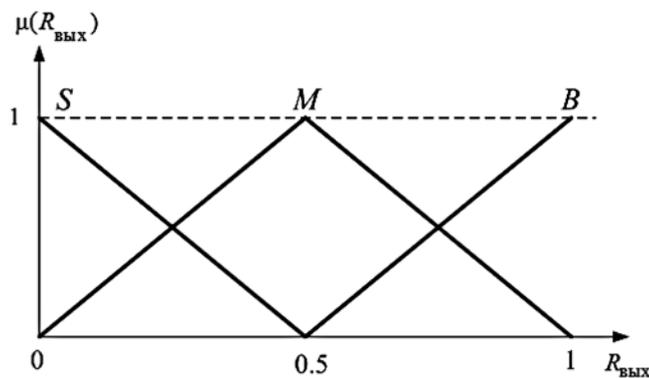


Рис. 9. Вид функций принадлежности терм-множества выходной лингвистической переменной с 3 термами

$\varepsilon = m_x, c = (m_x + 3\sigma_x), \gamma = (m_x + 6\sigma_x)$; из 7 термов:
 $\alpha = (m_x - 6\sigma_x), \beta = (m_x - 4\sigma_x), \theta = (m_x - 2\sigma_x), \varepsilon = m_x,$
 $\lambda = (m_x + 2\sigma_x), c = (m_x + 4\sigma_x), \gamma = (m_x + 6\sigma_x).$

Нечеткий вывод осуществлялся в соответствии с алгоритмом Мамдани для следующих условий:

- входной сигнал в первом канале находится в рабочем диапазоне, а во втором канале отклоняется от "математического ожидания" на $4\sigma_x, 5\sigma_x, 6\sigma_x$;
- входные сигналы в обоих каналах находятся за пределами рабочего диапазона.

Результаты машинного эксперимента представлены в таблице.

Результаты машинного эксперимента

Решение о состоянии системы (заключения)	Вес решений					
	для числа термов			для числа термов		
	3	5	7	3	5	7
	Отклонение в канале 2 на $4\sigma_x$			Отклонение в каналах 1, 2 на $4\sigma_x$		
I	0,586	0,472	0,474	0,586	0,472	0,472
II	0,178	0,178	0,169	0,413	0,472	0,472
III	0,413	0,527	0,526	0,413	0,472	0,472
IV	0,178	0,178	0,169	0,413	0,5	0,5
	Отклонение в канале 2 на $5\sigma_x$			Отклонение в каналах 1, 2 на $5\sigma_x$		
I	0,413	0,385	0,398	0,413	0,385	0,39
II	0,178	0,178	0,178	0,413	0,385	0,39
III	0,586	0,62	0,6	0,413	0,385	0,39
IV	0,178	0,178	0,178	0,586	0,575	0,552
	Отклонение в канале 2 на $6\sigma_x$			Отклонение в каналах 1, 2 на $6\sigma_x$		
I	0,163	0,163	0,163	0,163	0,163	0,163
II	0,163	0,163	0,163	0,163	0,163	0,163
III	0,836	0,836	0,836	0,163	0,163	0,163
IV	0,163	0,163	0,163	0,836	0,836	0,836

Из представленных в таблице значений видно, что при принятии решения о состоянии системы по заключению нечетких высказываний с максимальным весом терм-множества, имеющие от 3 до 7 термов, дают одинаковый результат при отклонении входной величины в диапазоне не менее $5\sigma_x$. Так, при отклонении в канале 2 сигнала на величину $5\sigma_x$ из таблицы видно, что максимальный вес для решения III о состоянии системы для числа терм от 3 до 7 практически одинаковый и составляет 0,586 для 3 термов, 0,62 для 5 термов и 0,6 для 7 термов. При отклонении менее $5\sigma_x$ наиболее рациональным является использование 5 термов. Аналогичная ситуация складывается при отклонении сигнала в каналах 1 и 2 на величину не менее 5, так максимальный вес для решения IV о состоянии системы составляет 0,586 для 3 термов, 0,575 для 5 термов, 0,552 для 7 термов.

Необходимо отметить, что увеличение числа термов приводит к увеличению правил нечеткого вывода $K_{ПР} = (K_{ТЕРМ})^s$, где s — число входных переменных, а $K_{ТЕРМ}$ — число термов.

Заключение

Экспериментальным путем показано, что нечеткая логика, используемая в резервируемых системах для обнаружения отказа резервируемого канала, может быть технически реализована на дешевой широко распространенной элементной базе, имеющей малые массогабаритные характеристики.

Минимальное число термов входной лингвистической переменной, позволяющей уверенно обнару-

живать отказы, при отклонении входной переменной не менее $5\sigma_x$ равно трем, а менее $5\sigma_x$ — пяти.

Список литературы

1. **Тарасов А. А.** Управление функциональной реконфигурацией отказоустойчивых систем // Информационные технологии. 2000. № 8. С. 11—15.
2. **Волобуев М. Ф., Демчук В. А., Замыслов М. А.** и др. Управление резервированной с помощью мажоритарных элементов системой с применением методов нечеткой логики // Информационные технологии. 2012. № 12.
3. **Соколов И. В., Мальцев А. М., Волобуев М. Ф.** и др. Оценка статистических характеристик сигнала на выходе резервированной системы, управляемой нечетким контроллером // Информационные технологии. 2014. № 8. С. 17—22.
4. **Волобуев М. Ф., Мальцев А. М., Соколов И. В.** и др. Способ резервирования систем с использованием методов нечеткой логики. Пат. ФИПС № 2565417 от 15.07.2015 г.

5. **Волобуев М. Ф., Мальцев А. М., Соколов И. В.** и др. Способ резервирования систем и устройство его реализации. Пат. ФИПС № 2604335 от 05.09.2016 г.

6. **Волобуев М. Ф., Замыслов М. А., Соколов И. В.** Способ экономического резервирования бортовых систем беспилотных летательных аппаратов на основе нечеткой логики // Академические Жуковские чтения. Современные проблемы и перспективные направления развития комплексов авиационного оборудования. Сб. статей по материалам II Всероссийской НПК. Воронеж: ВУНЦ ВВС "ВВА", 2015. С. 39—43.

7. **Леоненков А. В.** Нечеткое моделирование в среде MATLAB и fuzzyTECH. СПб.: БХВ-Петербург, 2005. 736 с.

8. **Антипов С. И., Дементьев Ю. В., Калинин А. Е.** Нечеткая логика и возможности ее применения в системах управления современного автомобиля // Материалы международной научно-технической конференции ААИ "Автомобиле- и тракторостроение в России: приоритеты развития и подготовки кадров", посвященный 145-летию МГТУ "МАМИ". М.: МАМИ, 2011. С. 11—20.

M. F. Volobuev, Doctoral Candidate, **A. M. Maltsev**, Senior Researcher,
I. V. Sokolov, Senior Researcher, Educator, e-mail: volmf81@mail.ru
Military Educational and Scientific Center of the Air Force
"N. E. Zhukovsky and Y. A. Gagarin Air Force Academy" (Voronezh)

An Experimental Study of the Layout of a Redundant System Controlled by Fuzzy Controller

A promising direction of improving the reliability of technical systems for various purposes is the redundancy of the most critical channels (blocks, devices) using controllers based on fuzzy multiple approach to detect failures and redundant control system, in accordance with which the selection and connection of one of the channels of the system to its exit. However, the implementation of the device failure detection using a fuzzy-set approach, based on the capabilities of modern microelectronic technology remained open. Also left open the question of the minimum number of terms a basic term-sets of an input linguistic variable, which allows to detect failures in redundant systems. The article considers the issues of practical implementation on a modern element base of devices to detect failures in redundant systems using fuzzy logic and a minimum number of terms of input linguistic variable, which allows to detect failures in redundant systems.

Keywords: layout, fuzzy logic, redundant system, therm

References

1. **Tarasov A. A.** Upravlenie funkcional'noj rekonfiguraciej otказоустойчивых систем, (Management of functional reconfiguration failure-resistant systems), *Informacionnye tehnologii*, 2000, no. 8, pp. 11—15 (in Russian).
2. **Volobuev M. F., Demchuk V. A., Zamyslov M. A., Mal'cev A. M., Mihajlenko S. B.** Upravlenie rezervirovannoj s pomoshh'ju mazhoritarnyh jelementov sistem s primeneniem metodov nechetkoj logiki (Management of redundant with the majority of elements of the system using methods of fuzzy logic), *Information technology*. 2012, no. 12 (in Russian).
3. **Sokolov I. V., Ageev A. M., Volobuev M. F., Zamyslov M. A., Mal'cev A. M., Mihajlenko S. B.** Ocenka statisticheskikh harakteristik signala na vyhode rezervirovannoj sistemy, upravljaemoj nechetkim kontrolleroom (Evaluation of the statistical characteristics of the signal at the output of the redundant system controlled by a fuzzy controller), *Information technology*, 2014, no. 8, pp. 17—22 (in Russian).
4. **Volobuev M. F., Zamyslov M. A., Sokolov I. V.** i dr. Sposob rezervirovanija sistem s ispol'zovaniem metodov nechetkoj logiki // Patent FIPS no.2565417 ot 15.07.2015 g. (The Way of backing up systems using methods of fuzzy logic), *Patent FIPS* no. 2565417 from 15.07.2015 g.) (in Russian).
5. **Volobuev M. F., Zamyslov M. A., Sokolov I. V.** i dr. Sposob rezervirovanija sistem i ustrojstvo ego realizacii (The Way of backup

systems and device for its implementation), *Patent FIPS* no. 2604335 from 05.09.2016 g.) (in Russian).

6. **Volobuev M. F., Zamyslov M. A., Sokolov I. V.** Sposob jekonomichnogo rezervirovanija bortovyh sistem bespilotnyh letatel'nyh apparatov na osnove nechetkoj logiki (The Method is economical redundancy of onboard systems of unmanned aerial vehicles based on fuzzy logic, Academic Zhukovskie reading. Current problems and promising directions of development of the complexes of aircraft equipment a Collection of articles on materials of the II all-Russian NPK.) *Akademicheskie Zhukovskie chtenija. Sovremennye problemy i perspektivnye napravlenija razvitiija kompleksov aviacionnogo oborudovanija [tekst] — Sbornik statej po materialam II Vserossijskoj NPK*: Voronezh, VUNC VVS "VVA", 2015, pp. 39—43 (in Russian).

7. **Leonenkov A. V.** *Nechetkoe modelirovanie v srede Matlab i fuzzyTech*, (Fuzzy modeling in Matlab and fuzzyTech), Saint Petersburg, "BHV-Peterburg", 2005. 250 p. (in Russian).

8. **Antipov S. I., Dement'ev Ju.V., Kalinin A. E.** *Nechetkaja logika i vozmozhnosti primenenija v sistemah upravlenija sovremennogo avtomobilja. Materialah mezhdunarodnoj nauchno-tehnicheskoi konferencii AAI "Avtomobile- i traktorostroenie v Rossii: prioritety razvitiija i podgotovki kadrov", posvjashhennyj 145-letiju MGTU "MAMI"*. (Fuzzy logic and its applicability in control systems of a modern car. Proceedings of international scientific and technical conference AAI "Car and tractor construction in Russia: development priorities and training", dedicated to the 145th anniversary of MGTU "MAMI"). Moscow, MAMI. 2011. pp. 11—20 (in Russian).

МОДЕЛИРОВАНИЕ И ОПТИМИЗАЦИЯ MODELING AND OPTIMIZATION

УДК 62-50; 519.7; 519.8

В. И. Левин, д-р техн. наук, проф., e-mail: vilevin@mail.ru,
Пензенский государственный технологический университет

Многомерные интервалы, их исчисление и применение

Приведена детальная разработка новой математической модели неопределенности — многомерного интервала, являющегося множеством конечного числа независимых интервалов неопределенности, системы алгебраических операций над многомерными интервалами и правил выполнения этих операций. Для выполнения поставленной цели предложено распространить на изучение многомерных интервалов известный в интервальной математике метод изучения интервалов, основанный на определении алгебраических операций над интервалами в виде теоретико-множественного обобщения соответствующих операций над вещественными числами. Новизна работы состоит в предложенной новой математической модели неопределенности систем в виде многомерных интервалов совместно с математическим аппаратом, позволяющим выполнять различные операции над многомерными интервалами и тем самым дающим возможность выполнять математическое моделирование систем с неопределенностью. Детально разработана новая математическая модель неопределенности — многомерный интервал, определены алгебраические операции над многомерными интервалами, выведены правила их выполнения. Предложен алгоритм изучения систем с многомерно-интервальными параметрами.

Ключевые слова: модель неопределенности, интервал, многомерный интервал, моделирование неопределенных систем

Введение

В последние десятилетия все чаще встречаются новые технологии, связанные с изучением неопределенности. Эти технологии широко применяют в технике, экономике и социальной сфере. Для их поддержки необходимы новые математические модели и методы. Поэтому данная работа, посвященная разработке новой модели неопределенности (многомерный интервал) и математических методов ее изучения, является актуальной.

Понятие неопределенности значительно расширилось, оно стало включать не только случайность возможных исходов, которая количественно изучается вероятностными методами, но и неединственность исходов или их незнание, дрейф переменных, семантическую неопределенность целей, многокритериальность принятия решений, недоопределенность модели или структуры исследуемой системы и т. д. Новые виды неопределенности изучаемых систем привели к разработке новых математических методов их изучения: теория нечетких множеств, многозначная логика, сверхслучайные процессы и др. Одним из самых популярных методов стала также интервальная математика, которая занимается изучением величин, определяемых с точностью до интервалов возможных значений [1, 2]. Но отдельные интервалы, изучаемые в интервальной математике, не могут охватить все случаи, встречающиеся на практике. Например, не-

определенная ситуация, характеризующая технологический процесс производства, может определяться набором независимых интервалов, каждый из которых определяет множество возможных значений температуры процесса, давления, скорости движущихся механизмов и т. д. В этом примере мы имеем неопределенные объекты нового вида — совокупности совместно рассматриваемых независимых интервалов, задающих области возможных значений параметров различной природы. Такие объекты естественно называть многомерными интервалами. Настоящая работа полностью посвящена теории и возможным применениям многомерных интервалов. Вместе с предыдущей работой автора [3], посвященной полиинтервалам, ее можно рассматривать как вклад в дальнейшее развитие интервальной математики.

1. Постановка задачи

Как подход к изучению неопределенных систем, интервальная математика строится на основе понятия интервала, рассматриваемого как множество всех возможных значений неполностью определенной величины \tilde{a} , задаваемой только ее нижней a_1 и верхней a_2 границами. Соответственно этому величина \tilde{a} записывается в форме ограниченного интервала неопределенности вида множества

$$\tilde{a} \equiv [a_1, a_2] = \{a \mid a_1 \leq a \leq a_2\}. \quad (1)$$

Здесь мы предполагаем, что неизвестное "истинное" значение неопределенной величины \tilde{a} достоверно лежит в пределах интервала $[a_1, a_2]$, не выходя за его границы a_1 и a_2 . Причем все значения в пределах этого интервала считаются равновероятными в том смысле, что нет никаких оснований предпочитать одно значение другому. Здесь понятие равновероятности не означает задание равномерного вероятностного или иного распределения возможных значений внутри данного интервала. Над интервалами вида (1) вводятся алгебраические операции, аналогичные соответствующим операциям над числами. Для этого используется следующая теоретико-множественная конструкция:

$$\tilde{a} \circ \tilde{b} = \{a \bullet b \mid a \in \tilde{a}, b \in \tilde{b}\}, \circ \tilde{a} = \{\bullet a \mid a \in \tilde{a}\}, (2)$$

т. е. любая операция над интервалами \circ определяется на основе соответствующей операции над точными величинами \bullet , при условии, что конкретные значения этих величин пробегает все возможные значения из соответствующих интервалов. Из определения (2) получаются следующие правила выполнения операций над интервалами:

$$\begin{aligned} [a_1, a_2] + [b_1, b_2] &= [a_1 + b_1, a_2 + b_2]; \\ [a_1, a_2] - [b_1, b_2] &= [a_1 - b_2, a_2 - b_1]; \\ k \cdot [a_1, a_2] &= \begin{cases} [ka_1, ka_2], k > 0, \\ [ka_2, ka_1], k < 0; \end{cases} \end{aligned} (3)$$

$$\begin{aligned} [a_1, a_2] \cdot [b_1, b_2] &= [\min_{i,j} (a_i \cdot b_j), \max_{i,j} (a_i \cdot b_j)]; \\ [a_1, a_2] / [b_1, b_2] &= [a_1, a_2] \cdot [1/b_2, 1/b_1] \text{ при } 0 \notin [b_1, b_2]. \end{aligned}$$

Продолжим развитие интервальной математики и введем понятие многомерного интервала как совокупности (множества) совместно рассматриваемых независимых интервалов

$$\tilde{M} = (\tilde{a}_1, \tilde{a}_2, \dots, \tilde{a}_p), (4)$$

где $\tilde{a}_1, \tilde{a}_2, \dots, \tilde{a}_p$ — интервалы вида (1), т. е. $\tilde{a}_i = [a_{i1}, a_{i2}]$.

Операции над многомерными интервалами определим аналогично операциям над интервалами, используя теоретико-множественные конструкции вида (2). Пусть \tilde{M} — многомерный интервал вида (4), имеющий размерность p , а

$$\tilde{N} = (\tilde{b}_1, \tilde{b}_2, \dots, \tilde{b}_p), (5)$$

где $\tilde{b}_1, \tilde{b}_2, \dots, \tilde{b}_p$ — интервалы вида (1), т. е. $\tilde{b}_i = [b_{i1}, b_{i2}]$, — другой многомерный интервал вида (4) той же размерности. Тогда операции над многомерными интервалами определяются в виде

$$\begin{aligned} \tilde{M} \circ \tilde{N} &= \{\tilde{a}_i \bullet \tilde{b}_i \mid \tilde{a}_i \in \tilde{M}, b_i \in \tilde{N}\}, \\ \circ \tilde{M} &= \{\bullet \tilde{a}_i \mid \tilde{a}_i \in \tilde{M}\}. \end{aligned} (6)$$

Здесь \tilde{M} — p -мерный интервал вида (4), а \tilde{N} — p -мерный интервал того же вида (5), но с другими составляющими его одиночными интервалами вида (1). Задача данной работы состоит в том, чтобы, исходя из определения операций над многомерными интервалами, вывести правила выполнения этих операций, аналогичные правилам (3) операций над интервалами.

2. Решение задачи

Формулы (6) показывают, что выполнение операций над многомерными интервалами может быть сведено к выполнению соответствующих операций над одиночными интервалами. Именно для выполнения операции \circ над парой многомерных интервалов \tilde{M}, \tilde{N} нужно выполнить соответствующую операцию \bullet для каждой пары \tilde{a}_i, \tilde{b}_i их одиночных интервалов с совпадающими номерами i и объединить результаты в единое множество, которое будет искомым результатом $\tilde{M} \circ \tilde{N}$. Аналогично, для выполнения операции \circ над одним многомерным интервалом \tilde{M} нужно выполнить соответствующую операцию \bullet для каждого его одиночного интервала \tilde{a}_i и объединить полученные результаты в единое множество, которое и даст искомым результат $\circ \tilde{M}$. Используя изложенный алгоритм, можно без труда установить правила для конструктивного выполнения различных операций над многомерными интервалами, имеющие вид формул, и по форме аналогичные правилам (3) выполнения операций над одиночными интервалами.

Начнем с установления формулы для конструктивного выполнения операции сложения многомерных интервалов. Для этого достаточно использовать изложенный выше алгоритм, учитывая, что в данном случае операция \circ есть сложение двух многомерных интервалов \tilde{M} и \tilde{N} , а операция \bullet есть сложение двух одиночных интервалов \tilde{a}_i, \tilde{b}_i , из которых состоят многомерные интервалы \tilde{M} и \tilde{N} (см. формулы (5), (6)). Это сложение выполняется по соответствующей формуле (3). В результате получим искомую формулу

$$\begin{aligned} \tilde{M} + \tilde{N} &\equiv \{\tilde{a}_1, \tilde{a}_2, \dots, \tilde{a}_p\} + \{\tilde{b}_1, \tilde{b}_2, \dots, \tilde{b}_p\} = \\ &= \{\tilde{a}_1 + \tilde{b}_1, \tilde{a}_2 + \tilde{b}_2, \dots, \tilde{a}_p + \tilde{b}_p\}, \end{aligned} (7)$$

где $\tilde{a}_i + \tilde{b}_i = [a_{i1} + b_{i2}, a_{i2} + b_{i2}]$, $i = \overline{1, p}$.

Аналогично устанавливаются формулы для конструктивного выполнения остальных операций над многомерными интервалами. Формула для конст-

руктивного выполнения операции вычитания многомерных интервалов имеет следующий вид:

$$\begin{aligned} \tilde{M} - \tilde{N} &\equiv \{\tilde{a}_1, \tilde{a}_2, \dots, \tilde{a}_p\} - \{\tilde{b}_1, \tilde{b}_2, \dots, \tilde{b}_p\} = \\ &= \{\tilde{a}_1 - \tilde{b}_1, \tilde{a}_2 - \tilde{b}_2, \dots, \tilde{a}_p - \tilde{b}_p\}, \end{aligned} \quad (8)$$

где $\tilde{a}_i - \tilde{b}_i = [a_{i1} - b_{i2}, a_{i2} - b_{i1}]$, $i = \overline{1, p}$.

Конструктивное выполнение операции умножения многомерного интервала на число имеет вид

$$k\tilde{M} \equiv k\{\tilde{a}_1, \tilde{a}_2, \dots, \tilde{a}_p\} = \{k\tilde{a}_1, k\tilde{a}_2, \dots, k\tilde{a}_p\}, \quad (9)$$

где $k\tilde{a}_i = \begin{cases} [ka_{i1}, ka_{i2}], k > 0, \\ [ka_{i2}, ka_{i1}], k < 0, \end{cases}$, $i = \overline{1, p}$.

Конструктивное выполнение операции умножения многомерных интервалов соответствует формулам

$$\begin{aligned} \tilde{M} \cdot \tilde{N} &\equiv \{\tilde{a}_1, \tilde{a}_2, \dots, \tilde{a}_p\} \cdot \{\tilde{b}_1, \tilde{b}_2, \dots, \tilde{b}_p\} = \\ &= \{\tilde{a}_1 \cdot \tilde{b}_1, \tilde{a}_2 \cdot \tilde{b}_2, \dots, \tilde{a}_p \cdot \tilde{b}_p\}, \end{aligned} \quad (10)$$

где $\tilde{a}_i \cdot \tilde{b}_i = [a_{i1}, a_{i2}] \cdot [b_{i1}, b_{i2}] = [\min_{r,s}(a_{ir} \cdot b_{is}), \max_{r,s}(a_{ir} \cdot b_{is})]$, $i = \overline{1, p}$, $r, s = 1, 2$.

Формула для конструктивного выполнения операции деления многомерных интервалов имеет вид

$$\begin{aligned} \tilde{M}/\tilde{N} &\equiv \{\tilde{a}_1, \tilde{a}_2, \dots, \tilde{a}_p\} / \{\tilde{b}_1, \tilde{b}_2, \dots, \tilde{b}_p\} = \\ &= \{\tilde{a}_1/\tilde{b}_1, \tilde{a}_2/\tilde{b}_2, \dots, \tilde{a}_p/\tilde{b}_p\}, \end{aligned} \quad (11)$$

где $\tilde{a}_i/\tilde{b}_i = [a_{i1}, a_{i2}] \cdot [1/b_{i2}, 1/b_{i1}]$, $i = \overline{1, p}$.

В формуле (11) выражение для частного двух интервалов \tilde{a}_i/\tilde{b}_i в виде произведения двух интервалов вычисляется по последней формуле в (10). Формулы (7)–(11) определяют правила конструктивного выполнения всех указанных выше алгебраических операций над многомерными интервалами путем сведения этих операций к соответствующим хорошо известным операциям над одиночными интервалами.

Теперь мы можем изложить алгоритм решения различных задач, связанных с исследованием систем, имеющих многомерные интервальные характеристики.

Шаг 1. Построение математической модели, представляющей решение задачи как вычисление и анализ некоторой функции \tilde{F} от аргументов — многомерных интервалов.

Шаг 2. Составление по построенной модели блок-схемы алгоритма вычисления и анализа функции \tilde{F} .

Шаг 3. Вычисление и анализ по составленной блок-схеме алгоритма многомерной интервальной

функции \tilde{F} , с использованием формул (7)–(11) выполнения различных операций над многомерными интервалами.

Пример. Работник служит в двух фирмах: *A* и *B*. В обеих фирмах его месячная заработная плата выплачивается в двух различных валютах — рублях и долларах США. При этом в фирме *A* она составляет $55\,000 \pm 1000$ рублей плюс 1000 ± 100 долларов, в фирме *B* — $60\,000 \pm 1500$ рублей плюс 1200 ± 100 долларов. Нужно определить суммарную месячную зарплату работника.

Решение. Шаг 1. В фирме *A* 1-ю зарплату работника можно представить в виде интервала $\tilde{a}_1 = [a_{11}, a_{12}] = [54\,000, 56\,000]$, а 2-ю зарплату — как интервал $\tilde{a}_2 = [a_{21}, a_{22}] = [900, 1100]$. Аналогично, в фирме *B* 1-ю зарплату работника можно представить в виде интервала $\tilde{b}_1 = [b_{11}, b_{12}] = [58\,500, 61\,500]$, а 2-ю зарплату — в виде интервала $\tilde{b}_2 = [b_{21}, b_{22}] = [1100, 1300]$. Тогда месячную зарплату работника в фирмах *A* и *B* можно представить соответственно следующими двумерными интервалами:

$$\tilde{M} = \{\tilde{a}_1, \tilde{a}_2\} = \{[54\,000, 56\,000], [900, 1100]\},$$

$$\tilde{N} = \{\tilde{b}_1, \tilde{b}_2\} = \{[58\,500, 61\,500], [1100, 1300]\}.$$

Суммарная месячная зарплата работника \tilde{P} равна сумме его месячных зарплат в фирмах *A* и *B*,

т. е. $\tilde{P} = \tilde{M} + \tilde{N}$. После подстановки значений \tilde{M} и \tilde{N} окончательно получаем

$$\begin{aligned} \tilde{P} &= \{\tilde{a}_1, \tilde{a}_2\} + \{\tilde{b}_1, \tilde{b}_2\} = \\ &= \{[54\,000, 56\,000], [900, 1100]\} + \\ &+ \{[58\,500, 61\,500], [1100, 1300]\}. \end{aligned}$$

Последняя формула и есть математическая модель решения задачи в виде вычисления суммы двух двумерных интервалов.

Шаг 2. Блок-схема алгоритма вычисления функции-модели, полученной на шаге 1 алгоритма, очевидна и содержит одну ступень, на которой определяется сумма двух двумерных интервалов.

Шаг 3. Вычислим двумерную интервальную функцию-модель, найденную ранее на шаге 1 алгоритма. Эта функция есть сумма двух двумерных интервалов. Применяя формулу (7) сложения p -мерных интервалов для случая $p = 2$, находим искомое значение

$$\begin{aligned} \tilde{P} &= \{\tilde{a}_1, \tilde{a}_2\} + \{\tilde{b}_1, \tilde{b}_2\} = \\ &= \{[54\,000, 56\,000], [900, 1100]\} + \\ &+ \{[58\,500, 61\,500], [1100, 1300]\} = \\ &= \{[112\,500, 117\,500], [2000, 2400]\}. \end{aligned}$$

Таким образом, суммарная месячная зарплата работника находится в интервалах $[112\,500, 117\,500]$ рублей и $[2000, 2400]$ долларов или, в иной форме записи, $115\,500 \pm 2500$ и 2200 ± 200 дол.

3. Обсуждение

Как мы видели выше, дальнейшее развитие концепции интервальной неопределенности [1] приводит к понятию многомерного интервала, характеризующего более сложную неопределенность, имеющую вид множества независимых интервалов неопределенности. Такая неопределенность характеризуется тем, что параметр системы не просто принимает какое-либо заранее неизвестное значение внутри заданного интервала, а имеется целое множество интервалов, внутри каждого из которых и принимают свои, заранее неизвестные значения соответствующие параметры системы. Эта более сложная модель неопределенности систем встречается достаточно часто в военном деле, технике, экономике и других областях и потому заслуживает изучения. Это изучение естественно проводить, используя подходы, имеющиеся в интервальной математике [1, 2], и продолжая их в направлении учета множественности интервалов. Подобно тому, как интервальная математика базируется на алгебре интервалов, математика многомерных интервалов базируется на алгебре таких интервалов. При этом, так же как в алгебре интервалов, в алгебре многомерных интервалов существуют простые зависимости между сложностью (т. е. длиной) операндов и сложностью результатов операции.

Заключение

В настоящей работе сформулирована задача изучения новой модели неопределенности — так называемого многомерного интервала, обобщающей известную модель неопределенности — интервал — на случай существования нескольких независимых, рассматриваемых совместно интервалов неопределенности. С помощью известной из интервальной математики теоретико-множественной конструкции, аналогично операциям над интервалами, введены операции над многомерными интервалами. Разработана методика сведения операций над многомерными интервалами к операциям над интервалами. С ее помощью выведены формулы для конструктивного выполнения всех операций над многомерными интервалами и построен соответствующий алгоритм. На примере из экономики проиллюстрирована практическая польза разработанной теории и методов.

Список литературы

1. Алефельд Г., Херцбергер Ю. Введение в интервальные вычисления. М.: Мир, 1987. 370 с.
2. Левин В. И. Интервальная математика и исследование систем в условиях неопределенности. Пенза: Изд-во Пензенского технологического ин-та, 1998. 55 с.
3. Левин В. И. Полиинтервалы, их исчисление и применение // Системы управления, связи и безопасности. 2016. № 3. С. 239—246.

V. I. Levin, Dr. of Tech Sci, Professor, e-mail: volevin@mail.ru,
Penza State Technological University

Multidimensional Intervals: Calculs and Applications

In recent decades there are more and more new technologies in the civilian and military spheres which associated with studying of uncertainty. These technologies are widely used in engineering, economics, social sphere. To support their new mathematical models and methods are needed. In this regard, this article dedicated to the development of new model of uncertainty (multidimensional interval) and mathematical methods of its study is relevant. The purpose of the article is in detailed design of a new adequate mathematical model of uncertainty — multidimensional interval, which is a set of finite number of independent intervals of uncertainty, the system of algebraic operations on multidimensional intervals and rules to perform these operations. To accomplish this goal we propose to extend to study multidimensional intervals the method from the interval mathematics based on the determination of algebraic operations on intervals in form of set-theoretic generalization of operations on real numbers. The novelty of the work lies in the proposed new mathematical model of uncertainty of systems in form of multidimensional intervals, together with mathematical tools allowing to perform various operations on multidimensional intervals and thereby enabling them to perform mathematical modeling of systems with uncertainty. The article detailed developed a new mathematical model of uncertainty — multidimensional interval. The algebraic operations on multidimensional intervals are determined and some rules for their implementation are output. The algorithm of study of uncertain systems with parameters in form of multidimensional intervals is presented.

Keywords: model of uncertainty, interval value, the multidimensional interval, modeling of systems with uncertainty

References

1. Alefel'd G., Hercberger Ju. *Vvedenie v interval'nye vychislenija*, Moscow, Mir, 1987, 370 p. (in Russian).

2. Levin V. I. *Interval'naja matematika i issledovanie sistem v uslovijah neopredelennosti*, Penza: Izd-vo Penzenskogo tehnologicheskogo in-ta, 1998, 55 p. (in Russian).

3. Levin V. I. *Poliinterval'y, ih ischislenie i primenenie*, *Sistemy upravlenija, svjazi i bezopasnosti*, 2016, no. 3, pp. 239—246 (in Russian).

Д. В. Титов, нач. управления, Газпром информ, D.Titov@inform.gazprom.ru,
И. В. Ретинская, д-р техн. наук, проф., **В. С. Ретинский**, д-р техн. наук, проф., e-mail: kto@migmil.ru,
 Российский государственный университет (национальный исследовательский университет)
 нефти и газа имени И. М. Губкина, г. Москва

Моделирование инфраструктуры центра сертификации

Рассмотрена организационная процедура выдачи сертификатов для доступа к служебной информации в НК Газпром. Разработаны способы представления информации из базы данных о сертификации в виде выборок, удобных для дальнейшего статистического анализа. Предложена система мониторинга, позволяющая динамически отслеживать загрузку операторов. Рассмотрены подходы к представлению инфраструктуры центра сертификации как системы массового обслуживания.

Ключевые слова: сертификат аутентификации, защита информации, теория массового обслуживания, моделирование, обработка данных, программирование, базы данных

Введение

В современных корпорациях, какова бы ни была их направленность, реализуется множество бизнес-процессов, как профильных, так и вспомогательных. Все они требуют хранения и обработки больших массивов информации. Возникающие проблемы и решения рассмотрим на примере ОАО "Газпром". В данной корпорации осуществляются профильные бизнес-процессы, характерные для нефтегазовой отрасли. К ним относятся добыча, переработка, хранение, транспортировка газа, газового конденсата, нефти и производство сопутствующей продукции. Кроме того, реализуются и многие десятки вспомогательных бизнес-процессов.

Для упомянутой выше и необходимой для обеспечения всех бизнес-процессов обработки больших массивов информации в ПАО "НК Газпром" образован единый центр обработки данных (ЦОД) с размещением на нескольких площадках в г. Москве. Виртуально все информационно-управляющие системы (ИУС), размещаемые в ЦОД, находятся в домене Gazprom.loc. Домен изолирован как от единой ведомственной сети передачи данных (ЕВСПД), так и от глобальной сети Интернет. Доступ пользователей осуществляется только посредством терминальных сессий. Отдельные роботизированные решения, реализованные на предприятиях, технически взаимодействуют с ЦОД на тех же принципах, что и пользователи — люди.

Такое решение потребовало разработки и внедрения специальных систем защиты информации, которая хранится и обрабатывается в ЦОД.

Было предложено использовать самое распространенное средство организации безопасного доступа пользователей к ресурсам ЦОД — построение частных защищенных виртуальных каналов (VPN) и организацию терминального доступа. Шифрование канала в этом случае выполняется на основе криптографических алгоритмов, построенных на так

называемых парах открытых и закрытых ключей (в РФ технология в целом называется инфраструктурой открытых ключей (ИОК)).

Для первичной аутентификации пользователей в ЦОД и генерации псевдослучайных последовательностей, на которых будет построено шифрование канала, используют признаки пользователя. Типичной реализацией средства хранения таких признаков является персональный ключевой носитель (например, eToken, RuToken и др.) с неотчуждаемой информацией (открытый и закрытый ключи и сертификат пользователя, содержащий необходимую информацию о нем).

Сертификат аутентификации в данном случае — это вся записываемая на носитель информация.

Организационный объем всех ИУС первого этапа предполагал, что число пользователей будет до 80 тысяч человек. А число устройств (серверов, межсетевых экранов и проч.), требовавших для работы сертификаты серверной аутентификации, — порядка нескольких тысяч.

Для организации терминального доступа используется оборудование *Citrix*, а для формирования собственно сессий — сертификаты аутентификации определенного вида. Функции обеспечения жизненного цикла сертификатов возложены на специально созданную *Инфраструктуру центров сертификации* домена Gazprom.loc (ИЦС). Инфраструктура построена на программном обеспечении *Microsoft*.

В данной работе приведены результаты исследования процесса обслуживания запросов на выдачу сертификатов аутентификации на основе данных, полученных на протяжении более трех лет.

Для этого решены следующие задачи:

- 1) проанализирована организационная структура ИЦС;
- 2) разработан способ представления данных о числе сертификатов и времени между их выпуском;

3) сделаны подходы к описанию работы ИЦС как системы массового обслуживания; проверены гипотезы, допускающие представление ИЦС как системы массового обслуживания (СМО) определенного типа с заявленными параметрами [1, 2].

1. Первичная обработка исходных данных для моделирования

Любое реальное моделирование сталкивается с проблемой адекватного представления исходных данных, правильной их трактовки с точки зрения конкретных типов моделей. В данном случае также пришлось преодолеть ряд трудностей. Это связано, прежде всего, с тем, что проектирование базы данных системы аутентификации изначально не предполагало получение выборок по времени обслуживания и интервалам между выдачей сертификатов, подсчет параметров этих выборок.

Администратором ИЦС домена Gazprom.loc были предложены несколько вариантов представления информации. Из них был выбран вариант выгрузки статистики по выпущенным сертификатам за весь период эксплуатации от каждого регионального центра сертификации.

На наш взгляд, это лучше всего соответствовало задаче построения модели СМО.

Стандартными средствами ИЦС была организована выгрузка информации в виде текстовых файлов, элементами которых являются табулированные строки (для каждого факта выпуска сертификата), каждая из которых может содержать до 80 позиций.

Однако в соответствии с Положением о коммерческой тайне в ОАО "Газпром", его дочерних обществах и организациях (ДО) большая часть данной информации является конфиденциальной. Поэтому для построения моделей было принято решение ограничиться минимально возможными для работы полями:

- номер операции в общей номенклатуре ИЦС;
- фамилия оператора, выпустившего сертификат;
- наименование предприятия пользователя;
- дата и время выпуска сертификата;
- наименование регионального центра сертификации.

Пример записи показан в таблице.

Для последующей обработки и моделирования были использованы файлы со статистической информацией от каждого регионального ЦС за период с 11.01.2011 — начала работы ИЦС до 5.05.2015. Таким образом, рассматриваемый период составил 1215 дней.

На рис. 1 показаны результаты первичной обработки выборок:

- число региональных центров сертификации, для каждого из них название и число сертификатов, выпущенных за рассматриваемый период;
- общее число операторов, хотя бы единожды выпустивших сертификат;
- общее число выпущенных сертификатов;
- время начала и окончания наблюдений;
- число предприятий, для работников которых были выпущены сертификаты (кроме контрагентов, для которых сертификаты выпускают в одной группе безопасности в отдельном ЦС).

Фрагмент выгрузки исходных данных ИЦС

№ п.п.	Наименование организации пользователя	Дата и время выпуска	Оператор регионального ЦС
436	Газпром трансгаз Ухта Users	09.05.2012 14:20	CNTR\AU.dryahlyh
437	Газпром трансгаз Ухта Users	09.05.2012 14:22	CNTR\AU.dryahlyh
438	Газпром трансгаз Ухта Users	09.05.2012 14:25	CNTR\AU.dryahlyh
439	Газпром трансгаз Ухта Users	09.05.2012 14:29	CNTR\AU.dryahlyh
441	Газпром ПХГ Невское УПХГ Users	09.06.2012 9:14	GAZPROM\nl.chernyadev
443	Газпром ПХГ Невское УПХГ Users	09.06.2012 9:17	GAZPROM\nl.chernyadev
444	Газпром ПХГ Невское УПХГ Users	09.06.2012 9:19	GAZPROM\nl.chernyadev
449	Газпром ПХГ Невское УПХГ Users	09.11.2012 10:36	GAZPROM\IO.Angarhaev
456	Газпром ПХГ Невское УПХГ Users	9/14/2012 2:45 PM	GAZPROM\IO.Angarhaev
457		9/15/2012 9:46 AM	GAZPROM\CA-IIS01\$
458		9/15/2012 9:51 AM	GAZPROM\CA-IIS02\$
459		9/17/2012 4:30 PM	CNTR\AU.dryahlyh
469	Газинформсервис Users	9/19/2012 9:23 AM	CNTR\AU.dryahlyh
470	Газинформсервис Users	9/19/2012 9:27 AM	CNTR\AU.dryahlyh
477	Газинформсервис Users	9/19/2012 9:43 AM	CNTR\AU.dryahlyh
478		9/24/2012 7:55 AM	CNTR\AU.dryahlyh
490	Газпром трансгаз Санкт-Петербург Users	9/25/2012 7:36 AM	CNTR\AU.dryahlyh
491	Газпром трансгаз Санкт-Петербург Users	9/25/2012 7:41 AM	CNTR\AU.dryahlyh

Примечания.

1. Строки, в которых не указана организация, относятся к сертификатам серверной аутентификации (не пользовательским).
2. В полях "дата", "время" представлен не стандартизированный формат данных.
3. Операторы ЦС записаны в виде: Региональный Центр Сертификации (РегЦС)/учетная запись оператора.

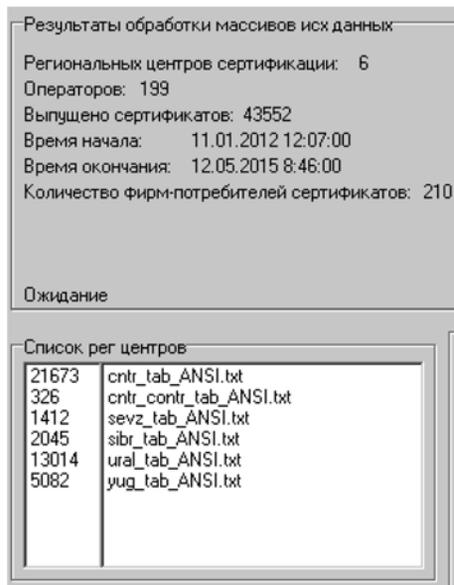


Рис. 1. Основные параметры выборки

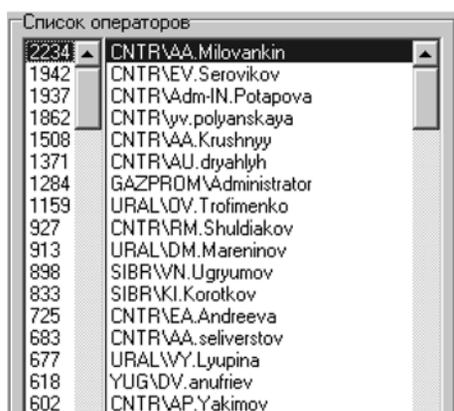


Рис. 2. Данные о числе сертификатов, выпущенных лидерами

Дальнейший анализ показал, что операторы выпустили существенно разное количество сертификатов. Лидеры — это сотрудники эксплуатирующей организации ООО "Газпром информ". Выпуск сертификатов — это основной вид их деятельности, и сертификаты они выпускают для всех предприятий, работающих в домене, — более 1000 шт. на человека (рис. 2). Аутсайдеры — операторы, выпустившие менее 10 сертификатов.

Небольшое число выпущенных оператором сертификатов необязательно говорит о его малом участии в работе ИЦС, возможны следующие варианты:

- оператор работал с ИЦС не на всем рассматриваемом периоде, а ограниченно по времени;
- число сертификатов для предприятия оператора мало, либо еще не начался массовый выпуск;
- оператор привлекается к работам эпизодически, только в моменты экстренного выпуска.

Среди предприятий, для которых выпускают сертификаты, также заметна большая разница в числе выпущенных сертификатов: от единичных

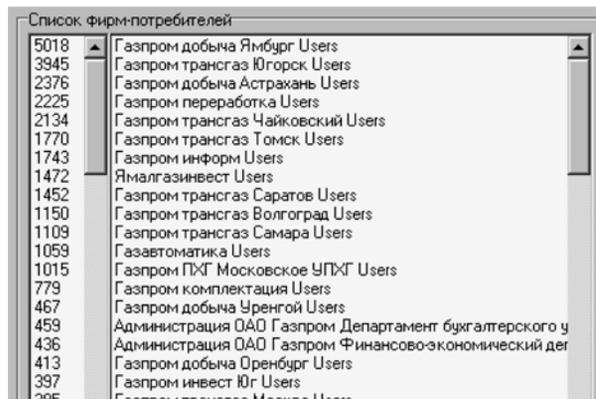


Рис. 3. Предприятия — лидеры

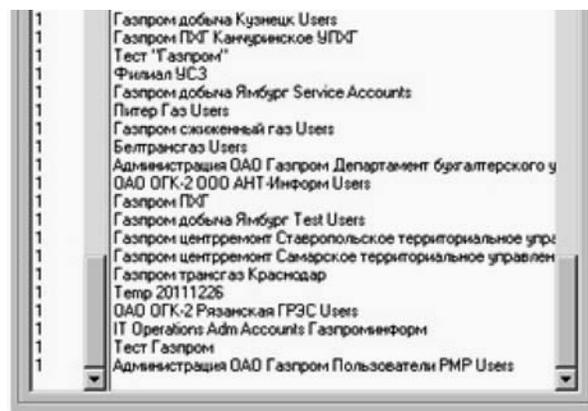


Рис. 4. Предприятия — аутсайдеры

до многотысячных. Лидеры, как видно по рис. 3, 4 — это предприятия, получившие свои ИУС.

Первичный анализ исходной информации позволил сделать выводы по некоторым параметрам работы ИЦС.

Итак, параметр *количество*:

- региональных центров сертификации фиксировано — шесть;
- число выпущенных сертификатов в региональных ЦС варьируется от 300 до 22 000. Это говорит о значительном расхождении в эффективности использования ЦС. Заметим, что на каждый региональный ЦС было выделено одинаковое количество вычислительных ресурсов и виртуального дискового пространства;
- операторов, выпускавших сертификаты, — 199 (фактически предоставлено около 350 полномочий на выпуск сертификатов);
- сертификатов, выпущенных одним оператором, — от 0 до 2300;
- число сертификатов, полученных предприятиями, колеблется от 1 до 5100.

Система мониторинга

Для дальнейшей обработки исходной информации был разработан программный продукт, пред-

ставляющий собой систему мониторинга загрузки операторов, в основные задачи которого входит:

1) интерпретация исходных данных и формирование из них выборки, графически и таблично отражающей деятельность по обеспечению сертификатами:

- региональных ЦС, ИЦС в целом;
- отдельных операторов и их групп;
- отдельных предприятий;

2) мониторинг работы операторов во времени;

3) получение выборок для проверки гипотез о виде вероятностного распределения и оценки параметров распределения, необходимых для дальнейшего моделирования ЦС как системы массового обслуживания (СМО) [1, 2].

Кроме того, для дальнейшего использования продукта реализованы следующие предельные значения по количеству:

- региональных центров сертификации — 10;
- операторов — 1000;
- предприятий — 500;
- сертификатов в ИЦС в целом — 500 000;
- сертификатов на одного оператора — 8000;
- сертификатов для одного предприятия — 10 000.

Также была обеспечена глубина оценки — 8 лет, из чего легко определить предельные объемы хранимой информации.

Программный продукт написан на языке программирования высокого уровня, удобном для изучения кода, а именно в среде разработки *Borland C++ Builder* версии 6.0. Программа структурирована и использует библиотеки визуальных компонентов для минимизации объема кода. Для реализации работы с визуальными компонентами использовали пакет визуализации VCL и некоторые компоненты из пакетов *Classes, Controls, StdCtrls, Forms, Dialogs, Chart, ExtCtrls, Series, TeEngine, TeeProcs, CGAUGES, ComCtrls*.

Работа с динамически создаваемыми массивами позволила избежать рисков исчерпания памяти при работе с большими объемами данных. Для этого используют объекты класса *TList*, позволяющие хранить произвольные массивы данных практически без ограничений по адресному пространству.

Результаты работы представляются в виде, удобном для трансляции в другие среды, и в виде масштабируемых графиков. Выгрузка текстовых и числовых данных осуществляется в объекты классов *TMemo, TListBox* с удобной навигацией по ним. Графики строятся в объектах класса *TChart* с полным набором средств визуализации и масштабирования. Динамическое изменение визуальных компонентов осуществляется посредством объекта класса *TTimer* с возможностью изменения частоты срабатывания с 10 до 1000 мс.

Общий алгоритм работы программы показан на рис. 5. Интерфейс программы рассчитан на работу по запросам оператора, т. е. после этапа подготовки



Рис. 5. Общая блок-схема программы

компонентов, инициализации массивов и глобальных переменных программа переходит в режим ожидания команд от пользователя. Команды отдаются посредством нажатия на соответствующие кнопки. Каждая команда вызывает функцию-обработчик.

Остановимся подробно на функции "Первичная обработка данных". Она предназначена для работы с динамическими массивами, полученными ранее. Эта работа состоит из следующих этапов:

- подсчет числа региональных ЦС;
- определение числа выпущенных сертификатов для каждого ЦС;
- поиск числа операторов;
- определение числа выпущенных сертификатов для каждого оператора;
- сортировка списка операторов по убыванию числа сертификатов;
- поиск числа предприятий пользователей;
- определение числа выпущенных сертификатов для каждого предприятия;
- сортировка списка предприятий по убыванию;
- определение времени начала и окончания наблюдений;
- вывод полученной информации на экран.

Функция "Обработка и отображение данных оператора" служит для определения времени и да-

ты выпуска сертификатов, которые, в свою очередь, позволяют определить:

- границы стационарного диапазона работы оператора;
- массив распределения времени поступления заявок;
- массив распределения времени между обработкой заявок;
- массив плотности заявок в день;
- массив плотности заявок в месяц.

По этим данным строят гистограммы распределений, которые затем используются для проверки гипотез о виде закона распределения.

Функция "Обработка и отображение данных предприятия" идентична предыдущей функции, за исключением того, что обрабатываются данные по наименованию фирмы, а не оператора.

С помощью функции "Запуск режима мониторинга" выполняется подготовка данных для пяти операторов (создание массивов выпуска сертификатов за рабочий день и глубины очереди для каждого из выбранных операторов с учетом заданного максимального числа выпускаемых сертификатов в день). Затем инициализируется объект класса *TTimer*, который обрабатывает с заданным периодом.

Каждое срабатывание таймера вызывает функцию — обработчик прерывания, в рамках которой происходит корректировка положения "ползунка" и перерисовка объектов отображения.

На рис. 6 и 7 (см. четвертую сторону обложки) показаны примеры работы программы.

Разработанное программное средство позволило наглядно показать многие аспекты работы ИЦС и проследить их динамику.

Например, анализ загрузки конкретных операторов показал, что практически все задействованные работники отдела криптографической защиты (ОКЗ) выполняют работы по выпуску сертификатов достаточно активно. На стационарных участках число выпускаемых ими сертификатов достигает тысячи и более в год. Ежедневное число сертификатов достигает 60 при работе в рабочее время и до 110 в авральном режиме.

Также активно выпускают сертификаты операторы тех дочерних организаций (ДО), для которых разворачиваются ИУС предприятий. Примером может служить ООО "Газпром добыча Ямбург", ООО "Газпром трансгаз Югорск" с более чем 5 и 3 тысячами сертификатов.

Всего 13 предприятий довели число пользователей, обеспеченных сертификатами, до более 500. Около 200 других ДО обеспечены сертификатами незначительно. При полной загрузке оператора достаточно от 1 до 10 рабочих дней (причем число ДО с менее чем 100 сертификатов составляет 2/3 от общего). Соответственно более 3/4 операторов либо вообще не выпускали сертификатов за рассматри-

ваемый период, либо занимались этой работой не более 1—2 рабочих дней.

На основании проведенного исследования может быть пока в общих чертах поставлена оптимизационная задача минимизации человеческих затрат на поддержку жизненного цикла сертификатов ИЦС с сохранением оперативности при многообразных ограничениях.

Визуализация процессов выпуска сертификатов — инструмент, позволяющий наглядно показать степень занятости операторов по времени, моменты наиболее активного выпуска сертификатов для конкретных предприятий. Возможен контроль производительности операторов, что является одной из важных функций аудита системы.

Подготовка к построению модели СМО

Примеры проверки гипотез

После формирования с помощью разработанной программы соответствующих выборок была проверена гипотеза H_0 о виде закона распределения времени обслуживания заявок. Это было необходимо для последующего представления ЦО как системы массового обслуживания. Был использован обычный χ^2 -критерий (ХИ-квадрат) с равновероятным разбиением. Проверялась гипотеза об экспоненциальном распределении генеральной совокупности времени обслуживания заявок и интервалов между поступлением заявок. В зависимости от длины ряда число интервалов изменялось для получения наилучшего результата. В данном случае гипотеза была сложной, так как параметр распределения оценивался по выборке. Значение статистики вычислялось по известной формуле

$$\chi^2 = N \sum_{i=1}^k \frac{\left(\frac{n_i}{N} - P_i(\theta)\right)^2}{P_i(\theta)}, \quad (1)$$

где $P_i(\theta)$ — теоретическая вероятность попадания наблюдения в i -й интервал при истинной гипотезе H_0 . Статистика в пределе подчиняется χ^2 -распределению с числом степеней свободы $s = k - p - 1$ для сложной гипотезы. Для принятия решения по гипотезе сравниваются значение статистики и критическое значение (квантиль) χ^2 -распределения (при заданном уровне значимости — обычно 0,05). Если значение статистики меньше значения квантиля, то гипотеза не отвергается, в противном случае — отвергается.

Примеры работы программы выглядят таким образом:

Для выборки времени между поступлением заявок:

$Xd1 := [36, 31, 12, 14, 7, 21, 6, 1, 6, 1, 1, 1, 5, 1, 1, 0, 2, 0, 0, 0, 2, 3, 0, 5, 1, 9, 7, 2, 3, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$

Значения статистики и квантиля:

$$XiE := 6,69181093721240$$

$$q := 7,81472828803626.$$

Следовательно, гипотеза об экспоненциальном распределении не отвергается.

Для выборки времени обслуживания заявок:

$$Xd1 := [969, 57, 17, 85, 14, 38, 30, 5, 32, 21, 4, 14, 11, 9, 12, 9, 2, 5, 8, 1, 6, 5, 1, 8, 2, 4, 2, 5, 4, 5, 1, 1, 4, 3, 2, 1, 4, 2, 1, 4, 4, 1, 3, 0, 1, 2, 2, 0, 2, 1]:$$

Значения статистики и квантиля:

$$XiE := 8,90222889434065$$

$$q := 9,48772903678116.$$

Следовательно, гипотеза об экспоненциальном распределении также не отвергается.

Кроме того, были вычислены оценки параметров распределения, в частности, оценки математического ожидания.

Полученные результаты служат основой для построения имитационной модели ЦС как системы массового обслуживания.

Глобальная и локальная модели ИЦС

Как известно из теории массового обслуживания [1, 2], во всякой СМО можно выделить следующие основные элементы:

- 1) входящий поток заявок;
- 2) очередь;
- 3) каналы обслуживания;
- 4) выходящий поток обслуженных заявок.

Изучение организационной структуры ИЦС с этой точки зрения дало возможность описать подходы к построению модели ИЦС как СМО.

На рис. 8 представлена схема ИЦС. Ее рассмотрение и изучение работы ИЦС в целом позволило установить следующее. Для ИЦС в целом (глобальная модель):

- входящий поток заявок на обслуживание — выпуск сертификатов;
- система многоканальная, имеются семь каналов обслуживания с бесконечными очередями (теоретически обрабатываются все поступающие заявки);
- выбор заявки происходит в зависимости от *порядка* ее поступления в очередь;
- время обслуживания одного клиента одинаковое;
- поток событий можно считать простейшим, т. е. обладающим свойствами стационарности, отсутствия последствия и ординарности. Это допущение конечно достаточно условно, однако проверка гипотез о законах распределения числа заявок и времени между заявками косвенно подтвердило этот факт;
- операторы Газпром информ могут работать на весь ЦС;

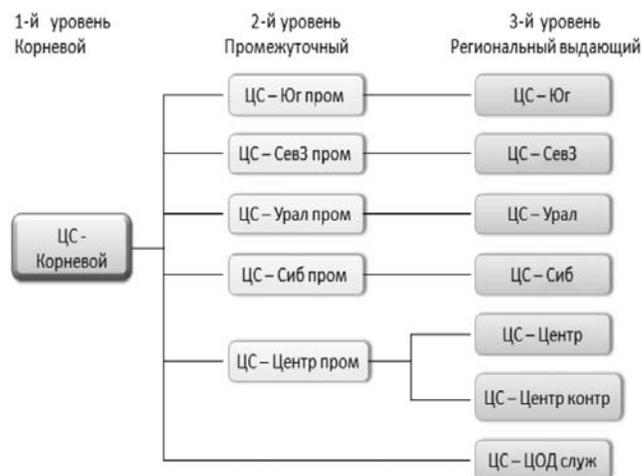


Рис. 8. Глобальная организационная структура ИЦС

- выходящий поток обслуженных заявок — сертификаты.

Для одного регионального центра сертификации (СМО локальная):

- входящий поток заявок на обслуживание — выпуск сертификата;
- число каналов обслуживания с бесконечными очередями произвольное и различное в разных ЦС;
- время обслуживания одного клиента одинаковое;
- каждый оператор обслуживает свою клиентуру (свое ДО);
- одно ДО может иметь нескольких операторов;
- только операторы Газпром информ могут работать на весь ЦС.

Как видно, получилась достаточно сложная модель. Пример схемы работы таков. При числе пользователей свыше 4000 человек равномерное поступление заявок на выпуск сертификатов невозможно. Заявка попадает к тому оператору, у которого наименьшая очередь. Если заявка поступает в момент превышения лимита загруженности оператора, задаваемого отдельно, заявка переходит к оператору Газпром информ.

Таким образом, может быть установлена зависимость между характером потока заявок, числом каналов обслуживания, производительностью отдельного канала и эффективным обслуживанием. В будущем это позволит определить такой вариант системы, при котором будет обеспечен минимум суммарных затрат от ожидания обслуживания, потеря времени и ресурсов на обслуживание и потеря от простоев каналов обслуживания.

Список литературы

1. **Карташевский В. Г.** Основы теории массового обслуживания. М.: Горячая линия—Телеком, 2013. 130 с.
2. **Матвеев В. Ф., Ушаков В. Г.** Системы массового обслуживания. М.: Изд-во МГУ, 1984. 127 с.

D. V. Titov, Head of Department, LLC "Gazprom inform" D.Titov@inform.gazprom.ru,
I. V. Retinskaya, DSc, Professor, **V. S. Retinsky**, DSc, Full Professor, kto@migmail.ru,
Gubkin Russian State University of Oil and Gas, Moscow

Modeling of the Certification Center Infrastructure

The organizational procedure for issuing certificates for access to official information in Gazprom corporation is considered. Methods for presenting information from the certification database in the form of samples convenient for further statistical analysis are developed. A monitoring system is proposed that allows you to dynamically monitor the loading of operators. Approaches to the presentation of the infrastructure of the Certification Center as a queuing system are considered.

Keywords: authentication certificate, information protection, queuing theory, modeling, data processing, programming, databases

References

1. **Kartashevskij V. G.** *Osnovy teorii massovogo obsluzhivaniya*. Moscow, Gorjachaja linija — Telekom, 2013, 130 p. [B. G. Kartashevskij. Fundamentals of queuing theory, Moscow, Hot line. Telekom, 2013, 130 p.]

2. **Matveev V. F., Ushakov V. G.** *Sistemy massovogo obsluzhivaniya*. Izdatel'stvo MGU, 1984, 127 p. [V. F. Matveev, V. G. Ushakov. Queuing systems. Publishing house of Moscow University, 1984, 127 p.]

УДК 681.518:519.876.2

В. Г. Литвин, д-р техн. наук, проф., гл. специалист, e-mail: litvg@mail.ru,
Научно-исследовательский институт автоматической аппаратуры
им. академика В. С. Семенихина (АО "НИИАА"), г. Москва,

Ю. В. Литвин, канд. экон. наук, директор центра, e-mail: litvinj@simplecs.ru,
Научно-исследовательский институт экономики и организации управления
в газовой промышленности (ООО "НИИГазэкономика", ОАО "Газпром"), г. Москва

Об оценке времени разработки крупномасштабной информационно-управляющей системы, создаваемой в структуре системы систем

Рассматривается новый подход к оценке времени разработки проекта информационно-управляемой системы в организационной структуре системы систем в соответствии с волновой моделью и технологией виртуального прототипирования, концептуального и комбинированного моделирования. Времена параллельного выполнения работ оцениваются методами порядковых статистик при различных исходных распределениях. Использование методов продемонстрировано примерами расчета.

Ключевые слова: информационно-управляющие системы, волновая модель проекта, система систем, фазовые распределения

Введение

Небывалый рост и сложность создаваемых информационно-управляющих систем (ИУС) сопровождается развитием существующих и появлением новых методов их анализа, позволяющих адекватно оценивать и отвечать на возникающие вызовы информатизации. Одним из набирающих популярность подходов развития общей теории систем является создание понятийного аппарата, методологии и методов анализа так называемой системы систем (System of Systems — SOS) [1]. В данной

работе под SOS понимается совокупность независимых организаций-субподрядчиков, осуществляющих проектирование и разработку ИУС. Процесс проектирования, как правило, является многостадийным с наращиванием знаний о будущей системе и ее окружении. Решения, принятые в начале проекта, по мере продвижения по отдельным стадиям уточняются и дополняются. Для управления разработкой проектов предложен ряд моделей, наиболее часто применяемыми среди которых являются каскадная модель, V-образная модель, спиральная модель и др. [2]. В случае проектов, реализуемых в

структуре SOS, используется так называемая волновая модель [3].

Организации, входящие в SOS и участвующие в проектировании создаваемой системы, ведут разработку в соответствии с принятой в рамках проекта технологии. Именно уровень компетенций в соответствующих областях бизнеса и возможность использования принятой технологии разработки могут оказаться главными при выборе субподрядчиков для реализации сложных ИУС. Одним из вариантов такой технологии является совместное использование трех направлений разработки и анализа [4]: 1) концептуальное моделирование; 2) виртуальное прототипирование; 3) комбинированное имитационно-аналитическое моделирование. Использование виртуального прототипирования и моделирования позволяет не только отработать проектные решения по архитектуре, реализуемым функциям и интерфейсам, но и получить более или менее объективную информацию о результативности работы подсистем после принятых решений. Так как создание сложной системы в среде SOS субподрядчики выполняют в основном параллельно, то существенную роль при расчете времени выполнения всех работ играют модели оценки вероятностно-временных характеристик параллельно развивающихся случайных процессов [5, 6].

В настоящей работе решается задача определения времени разработки ИУС, реализуемой в структуре SOS. При этом путем аппроксимации исходных распределений времени выполнения отдельных фрагментов и подсистем и использования модельной оценки параллельно выполняемых работ необходимо получить итоговое распределение времени разработки ИУС и оценить по нему усредненные характеристики проекта и риски возможных превышений установленных планом сроков.

Волновая модель организации работ по проекту в структуре SOS

Территориально распределенная структура разработки ИУС возникает всякий раз, когда ведется

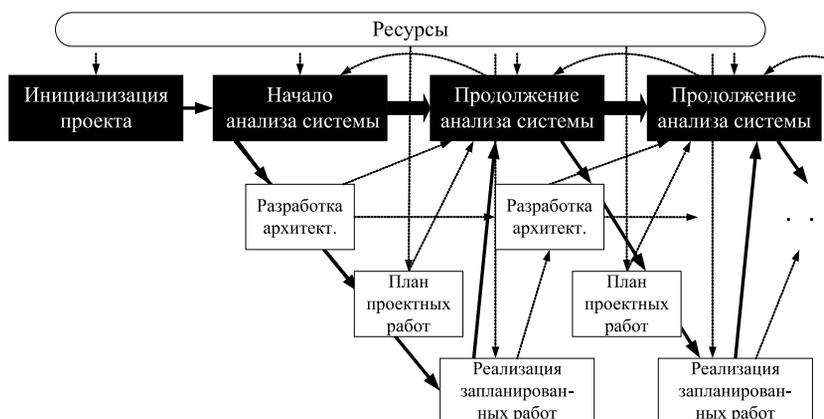


Рис. 1. Волновая модель разработки ИУС

автоматизация процессов управления различными бизнесами в рамках одной системы. Учет специфики того или иного бизнеса требует соответствующей специализации и компетенций разработчиков, что является причиной привлечения различных организаций для реализации проекта. Возникающее временное объединение организаций (субподрядчиков) характеризуется особенностями, присущими SOS. Отметим некоторые из них, которые принимаются во внимание при организации выполнения подобных проектов [1].

1. Операционная независимость входящих в SOS организаций (субподрядчиков). Эти организации имеют собственные цели бизнеса и способны работать и выполнять свои задачи даже после исключения их из возникшей структуры SOS.

2. Организационная независимость организаций, имеющих свою структуру, систему управления и цели бизнеса.

3. Входящие в SOS организации сами по себе являются сложными системами, неоднородными в организации, по направлениям деятельности и находятся на различных фазах своего жизненного цикла. Их включение в состав исполнителей проекта продиктовано спецификой задач и возможностями разработки.

Существует ряд других свойств SOS, которые здесь не приводятся, так как они не будут использованы в оценках процессов управления проектами создания ИУС.

Как отмечалось выше, процесс проектирования и разработки ИУС, как и других систем, обычно делят на стадии, на каждой из которых принимаются и реализуются те или иные решения. Для анализа SOS и систем, разрабатываемых в ее структурах, чаще всего используют волновую итерационную модель (рис. 1). В течение каждой стадии ("волны") выполняют все намеченные для нее работы, согласуются и оцениваются ранее принятые решения, ставятся вопросы, требующие проработки на следующей стадии. Например, работы, относящиеся к начальному периоду реализации проекта, выполняются в соответствии с методикой [7], базирующейся на виртуальном прототипировании, концептуальном и комбинированном моделировании. Концептуальное моделирование часто ведется с использованием средств UML и методов иерархического многокритериального анализа процессов [8]. Оно включает декомпозицию ИУС, обоснование требований к системе и ее компонентам, содержательную проработку архитектурных и других вопросов. Далее субподрядчики разрабатывают виртуальные прототипы каждой подсистемы, реализующие намеченные для них основные функции, и ведут согласование внутренних и внешних интерфейсов. Использование про-

тотипов создаваемых подсистем позволяет уже на ранних этапах подключить заказчика к определению детальных требований к создаваемой системе, а разработчикам провести апробирование принятых конструкторских решений.

Кроме того, использование виртуальных прототипов позволяет собрать данные о вычислительном процессе, оценить правильность принятых решений и обеспечить исходными данными последующее моделирование. Таким образом, полученные оценки на очередной стадии разработки ИУС используются затем для уточнения архитектурных и других решений и условий перехода к следующей стадии.

Для прогноза времени выполнения проекта в целом требуется определение длительности каждой стадии с последующей полной оценкой проекта. Если же оказывается, что планируемые сроки проекта нарушаются, то следует вернуться к уже выполненным стадиям и попытаться принять решения, реализация которых устранил проблему.

Оценка времени создания ИУС в структуре SOS

Представим граф, отражающий структуру волновой модели, в виде последовательности стадий, каждая из которых содержит параллельные узлы (рис. 2). Узел 0 обозначает центр проработки и принятия общесистемных решений и координации выполнения проекта, а другие узлы обозначают субподрядчиков, входящих в SOS. Время выполнения работ центром определяется узлами А и Ф. Время разработки подсистем каждым субподрядчиком связано с остальными узлами графа. На каждой стадии в выполнении проекта может участвовать различное число субподрядчиков, что и отражено соответствующим числом узлов N_i , $i = \overline{1, K}$, где K — число стадий.

Вопросам оценки вероятностно-временных характеристик разработки проектов, с которыми работают менеджеры, посвящено множество исследований. В настоящей работе ввиду параллельной работы субподрядчиков, входящих в SOS, данная проблема решается путем применения методов порядковых статистик [5]. Использование данного

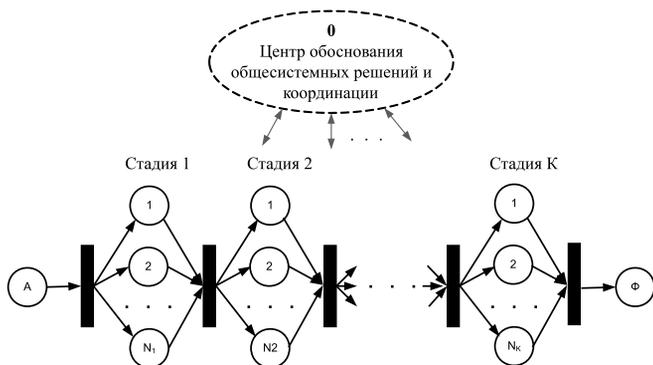


Рис. 2. Граф, отражающий верхний уровень волновой модели

аппарата совместно с методами свертки распределений позволяет при различных сценариях реализации проектов интегрировать процессы их выполнения и оценивать показатели эффективности и риски нарушения требований и планов. Аппроксимация распределений исходных случайных величин, характеризующих времена выполнения работ отдельными структурами, существенно расширяет возможности явного описания результирующих процессов. Особое место среди таких подходов занимают методы фазовых распределений (PH-распределений) [9, 10]. Фазовое распределение можно интерпретировать следующим образом. Представим гипотетическую систему обслуживания, в которую поступает одна заявка. Попав в систему, заявка может проходить через n фаз, в каждой из которых она задерживается на экспоненциальное время (экспоненциальное распределение с параметром λ_i , $i = \overline{1, n}$). По завершению фазы i заявка либо с вероятностью θ_{ij} попадает на новую фазу j , либо поглощается состоянием $n + 1$ с вероятностью θ_{in+1} ; $\sum_{j=1}^{n+1} \theta_{ij} = 1$. В каж-

дый момент в системе находится не более одной заявки. Время пребывания заявки в системе (до момента поглощения) будет иметь фазовое распределение. Выпишем явные матричные формулы для PH-распределения n -го порядка.

Распределение случайной величины фазового типа определяется процессом нахождения заявки в системе обслуживания $X(t)$ через инфинитезимальный оператор Q :

$$Q = \begin{pmatrix} T & T^0 \\ 0 & 0 \end{pmatrix},$$

где T — $n \times n$ матрица интенсивностей — транзитных переходов процесса $X(t)$; T^0 — вектор-столбец интенсивностей с n компонентами, определяющий переходы процесса $X(t)$ из транзитных состояний в поглощающее.

Функция и плотность распределения (ФР и ПР) случайной величины фазового типа будут иметь следующий вид [9]:

$$F(t) = 1 - \alpha \exp(Tt)e, \quad t \geq 0;$$

$$f(t) = \alpha \exp(Tt)T^0, \quad t \geq 0,$$

где e — вектор-столбец с n компонентами, равными единице; $\alpha^T = (\alpha_1, \alpha_2, \dots, \alpha_n)$ — вектор вероятностей поступления заявки в систему обслуживания на фазу, с которой начинается ее движение по остальным фазам ($\alpha^T \times e = 1$).

Преобразование Лапласа ПР $f(t)$ запишется следующим образом:

$$\tilde{f}(s) = \alpha(sI - T)^{-1}T^0, \quad \text{Re}(s) \geq 0, \quad (1)$$

где I — единичная квадратная матрица размерности.

Из выражения (1) путем многократного дифференцирования, полагая, что $s \rightarrow 0$, можно получить выражения для начальных моментов (m_k) *РН*-распределения:

$$m_k = (-1)^k k! \alpha \mathbf{T}^{-k} \mathbf{e}.$$

На практике чаще всего интересуются моментами первого и второго порядка — математическим ожиданием (m_1), дисперсией (σ^2), средним квадратическим отклонением (σ), коэффициентом вариации (C), а также коэффициентом корреляции (ρ). Для наших целей, когда рассматриваются независимые случайные величины, будем использовать m_1 , σ и C . Приведем их выражения для *РН*-распределения:

$$m_1 = -\alpha \mathbf{T}^{-1} \mathbf{e};$$

$$\sigma^2 = m_2 - m_1^2 = 2\alpha \mathbf{T}^{-2} \mathbf{e} - (\alpha \mathbf{T}^{-1} \mathbf{e})^2;$$

$$C = \frac{\sigma}{m_1}.$$

Существует множество примеров случайных величин с фазовыми распределениями: экспоненциальное и пуассоновское распределение; распределения Эрланга и гиперэрланга; распределение Кокса и др. Для иллюстрации приведем пример эрланговского распределения, как частного случая *РН*-распределения. В этом случае матрица \mathbf{T} и вектор \mathbf{T}^0 будут иметь следующий вид:

$$\mathbf{T} = \begin{pmatrix} -\lambda_1 & \lambda_1 & 0 & \dots & 0 \\ 0 & -\lambda_2 & \lambda_2 & \dots & 0 \\ 0 & 0 & -\lambda_3 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & -\lambda_n \end{pmatrix}, \mathbf{T}^0 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \dots \\ \lambda_n \end{pmatrix}.$$

Преобразование Лапласа плотности распределения $f(t)$ и моменты этого распределения будут иметь следующий вид:

$$\tilde{f}(s) = \prod_{i=1}^n \frac{\lambda_i}{s + \lambda_i}; m_1 = \sum_{i=1}^n \frac{1}{\lambda_i}; \sigma^2 = \sum_{i=1}^n \frac{1}{\lambda_i^2};$$

$$C^2 = \left(\sum_{i=1}^n \frac{1}{\lambda_i^2} \right) / \left(\sum_{i=1}^n \frac{1}{\lambda_i} \right)^2,$$

$C^2 \leq 1$, так как для неотрицательных действительных чисел a_i выполняется неравенство $\sum_i a_i^2 \leq \left(\sum_i a_i \right)^2$.

Для произвольных распределений времени выполнения работ отдельными субподрядчиками, включая случаи с $C^2 \geq 1$, можно воспользоваться известной аппроксимацией Кокса [11].

Так как основной особенностью разработки проекта в структуре SOS является параллельное

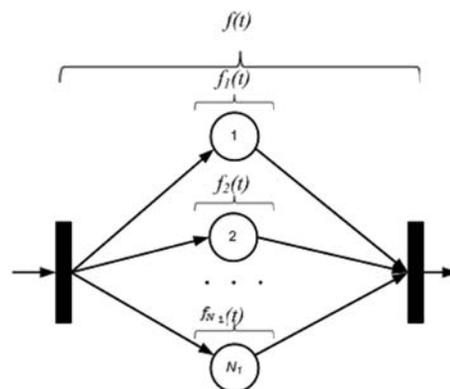


Рис. 3. Параллельное выполнение работ на стадии 1

проектирование отдельных подсистем, то остановимся кратко на явных выражениях для ПР времени параллельного выполнения отдельной стадии.

Рассмотрим фрагмент графа (рис. 3) стадии 1 с N_1 субподрядчиками, параллельно ведущими разработку системы. Каждая организация разрабатывает свою подсистему независимо от других со своими ФР $F_i(t)$ и ПР $f_i(t)$, $i = \overline{1, N_1}$, что характерно для организаций, входящих в SOS. Моменты времени завершения выполнения работ обозначим через X_1, X_2, \dots, X_{N_1} . Упорядочим их по возрастанию, введя новое обозначение $X_{(r)}$, указывающее на r -е по порядку завершение одной из работ. Тогда упорядоченная последовательность времен запишется так: $X_{(1)} \leq X_{(2)} \leq \dots \leq X_{(N_1)}$.

Понятно, что ввиду случайной длительности выполнения каждой работы i , на r -й позиции может оказаться любая из них. Тогда плотность распределения интервала от начала выполнения работ до момента $X_{(r)}$ будет иметь следующий вид [5, 6]:

$$f_{(r)}(t) = \frac{1}{(r-1)!(k-r)!} \sum_P F_{i_1}(t) \dots F_{i_{r-1}}(t) f_{i_r}(t) \times [1 - F_{i_{r+1}}(t)] \dots [1 - F_{i_k}(t)], \quad (2)$$

где \sum_P — обозначает сумму всех $k!$ перестановок (i_1, i_2, \dots, i_k) из $\overline{1, k}$; $t \in (0, \infty)$.

Выражение (2) может быть представлено через перманент в матричной форме, что снижает громоздкость записи, а в некоторых случаях и упрощает процесс вычислений [5]:

$$f_{(r)}(t) = \frac{1}{(r-1)!(k-r)!} \text{Per}(\mathbf{A}_r), \quad (3)$$

где $\text{Per}(\mathbf{A})$ — перманент матрицы \mathbf{A} ;

$$(\mathbf{A}_r) = \left(\begin{array}{cccc} F_1(t) & F_2(t) & \dots & F_k(t) \\ f_1(t) & f_2(t) & \dots & f_k(t) \\ 1 - F_1(t) & 1 - F_2(t) & \dots & 1 - F_k(t) \end{array} \right) \begin{array}{l} \} r - 1 \\ \} 1 \\ \} k - 1 \end{array}, \quad (4)$$

где $\}i$ — обозначает i совпадающих строк матрицы.

Перманент определяется подобно детерминанту матрицы, только все слагаемые при его раскрытии берутся с положительным знаком. Существуют алгоритмы, облегчающие его расчет [12]. Подобные по структуре формулы используют и для дискретных распределений времени выполнения работ.

Получив приближенные выражения для ФП и ПР времени выполнения каждой стадии проекта, посредством операций свертки можно определить вероятностные распределения длительности выполнения проекта в целом. По полученному распределению можно оценивать усредненные показатели времени выполнения проекта и различные типы квантилей и рисков.

Примеры оценки распределения времени выполнения проекта в структуре SOS

Пример 1. Рассмотрим упрощенный пример расчета ПР времени выполнения проекта в течение одной стадии. Будем считать, что для выполнения проекта привлечена организация-подрядчик (центральное звено SOS) и три организации-субподрядчики, которые будут выполнять разработку прикладных программ проекта в течение этой стадии. Таким образом, SOS состоит из четырех организаций. Укрупненный граф для рассматриваемого случая представлен на рис. 4. Узел А графа соответствует работе головной организации по проведению предварительных оценок выполнения работ: декомпозиция проекта; определение требований к функциям, интерфейсам; временам выполнения отдельных фаз и затратам. Состояния S_1 , S_2 и S_3 обозначают субподрядчиков, параллельно выполняющих свои работы. В течение фазы Ф готовится документация и проводятся комплексные испытания разработанной системы.

Как следует из графа, для определения ПР $f(t)$ времени выполнения проекта необходимо вначале найти распределение параллельной фазы $f_S(t)$, выполняемой субподрядчиками, а затем методами свертки найти искомое распределение. Для иллюст-

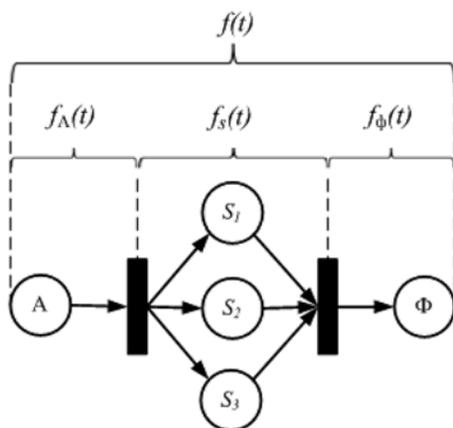


Рис. 4. Граф состояний

рации алгоритма вычисления ПР времени разработки проекта выпишем основные зависимости. В рассматриваемом случае матрица (4) будет иметь вид

$$A_3(t) = \begin{pmatrix} F_1(t) & F_2(t) & F_3(t) \\ F_1(t) & F_2(t) & F_3(t) \\ f_1(t) & f_2(t) & f_3(t) \end{pmatrix}.$$

Используя выражения (2) и (3), получим

$$f(t) = f_{(3)}(t) = 0,5 \text{Per}(A_3) = F_1(t)F_3(t)f_2(t) + F_2(t)F_1(t)f_3(t) + F_3(t)F_2(t)f_1(t),$$

где в $f_{(3)}(t)$ — индекс 3 указывает на распределение максимального времени параллельной работы тремя организациями.

Ввиду частого использования на практике экспоненциальных приближений и простоты расчетов рассмотрим случай параллельной работы субподряд-

чиков с ПР $f_i(t) = \lambda_i e^{-\lambda_i t}$, $i = \overline{1, 3}$. Преобразование Лапласа для ПР максимального времени параллельного выполнения работ будет следующим [6]:

$$\tilde{f}_{(3)}(s) = \sum_{i=1}^3 \frac{\lambda_i}{s + \lambda_i} - \frac{\lambda_1 + \lambda_2}{s + \lambda_1 + \lambda_2} - \frac{\lambda_1 + \lambda_3}{s + \lambda_1 + \lambda_3} - \frac{\lambda_2 + \lambda_3}{s + \lambda_2 + \lambda_3} + \frac{\lambda_1 + \lambda_2 + \lambda_3}{s + \lambda_1 + \lambda_2 + \lambda_3}.$$

Тогда начальные моменты из $\tilde{f}_{(3)}(s)$ определяются известным образом:

$$m_{1(3)} = -\frac{d}{ds} \tilde{f}_{(3)}(s)_{s=0} = \sum_{i=1}^3 \frac{1}{\lambda_i} - \frac{1}{\lambda_1 + \lambda_2} - \frac{1}{\lambda_1 + \lambda_3} - \frac{1}{\lambda_2 + \lambda_3} + \frac{1}{\lambda_1 + \lambda_2 + \lambda_3};$$

$$m_{2(3)} = \frac{d^2}{ds^2} \tilde{f}_{(3)}(s)_{s=0} = 2 \left[\sum_{i=1}^3 \frac{1}{\lambda_i^2} - \frac{1}{(\lambda_1 + \lambda_2)^2} - \frac{1}{(\lambda_1 + \lambda_3)^2} - \frac{1}{(\lambda_2 + \lambda_3)^2} + \frac{1}{(\lambda_1 + \lambda_2 + \lambda_3)^2} \right],$$

где $m_{1(3)}$ и $m_{2(3)}$ — первый и второй начальные моменты случайного времени параллельного выполнения работ для рассматриваемой стадии.

Через найденные начальные моменты легко определить среднее квадратическое отклонение и коэффициент вариации искомого времени.

Пример 2. Рассмотрим численный пример для случая, когда плотности распределений примера 1 заданы на дискретном множестве. Пусть X принимает целочисленные значения времени $(0, 1, 2, \dots, N)$ с вероятностями $f(k) = P(t = k)$ для $k \in \overline{(0, N)}$. Если $X = k$, то это будет значить, что время разработки

проекта составит k месяцев. Зададим в табл. 1 конкретные значения для графа рис. 4 (данные условные).

Запишем формулу расчета параллельного выполнения работ для двух субподрядчиков:

$$f_{S_1 S_2}(k) = f_{S_1}(k) \sum_{i=0}^k f_{S_2}(i) + f_{S_2}(k) \sum_{j=0}^{k-1} f_{S_1}(j). \quad (5)$$

Далее, воспользовавшись полученным результатом (5), определим свертку распределений $f_{S_1 S_2}(k)$ и $f_{S_3}(k)$.

В результате найдем распределение максимального времени параллельной разработки программ тремя субподрядчиками на рассматриваемой стадии $f_{S_1 S_2 S_3}(k)$ (табл. 2):

$$f_{S_1 S_2 S_3}(k) = f_{S_1 S_2}(k) \sum_{i=0}^k f_{S_3}(i) + f_{S_3}(k) \sum_{j=0}^{k-1} f_{S_1 S_2}(j).$$

Для определения ПР времени выполнения проекта воспользуемся формулой свертки для распределений $f_A(k)$, $f_{S_1 S_2 S_3}(k)$ и $f_\Phi(k)$:

$$f(k) = f_A(k) * f_{S_1 S_2 S_3}(k) * f_\Phi(k),$$

где $*$ — операция свертки.

Таблица 1

Распределение времени выполнения работ отдельными субподрядчиками

Наименование переменных	Распределение времени выполнения			
k , мес	2	3	4	
$f_A(k)$	0,1	0,5	0,4	
k , мес	3	5	6	
$f_\Phi(k)$	0,2	0,6	0,2	
k , мес	6	9	10	12
$f_{S_1}(k)$	0,1	0,3	0,4	0,2
k , мес	7	8	9	10
$f_{S_2}(k)$	0,1	0,8	0,1	
k , мес	6	8	11	12
$f_{S_3}(k)$	0,2	0,3	0,3	0,2

Таблица 2

Распределение времени параллельного выполнения работ тремя субподрядчиками

Наименование переменных	Распределение времени выполнения					
k , мес	7	8	9	10	11	12
$f_{S_1 S_2 S_3}(k)$	0,002	0,043	0,155	0,2	0,24	0,36

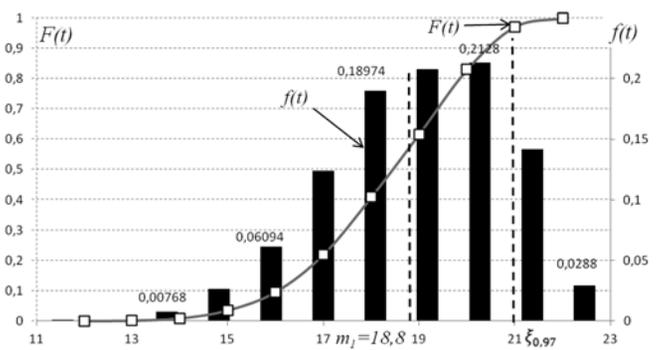


Рис. 5. Распределение времени выполнения проекта

Для дискретных случайных величин часто удобно пользоваться моментной производящей функцией (МПФ). В качестве примера выпишем ее только для случайной величины X_A с распределением $f_A(k)$:

$$M_{X_A}(t) = E(e^{tX_A}) = 0,1e^{-2t} + 0,5e^{-5t} + 0,4e^{-6t}.$$

Остальные МПФ выписываются аналогичным образом. Моментная производящая функция для времени выполнения проекта будет иметь следующий вид (приводится без детальной расшифровки):

$$M(t) = M_{X_A}(t) M_{X_S}(t) M_{X_\Phi}(t) = E(e^{t(X_A + X_S + X_\Phi)}),$$

где X_A , X_S и X_Φ — времена выполнения работ узлами A , S и Φ .

Результаты расчета распределения времени выполнения проекта приведены на рис. 5.

Среднее время выполнения проекта составило 18,8 мес, среднее квадратическое отклонение равно 1,7 мес. Для рассматриваемой дискретной случайной величины X квантиль ξ_p определяется с помощью следующего неравенства:

$$P(X < \xi_p) \leq p \leq P(X \leq \xi_p).$$

Для $\xi_p = 21$, $p = 0,97$.

Таким образом, организация проектных работ в структуре SOS обеспечивает независимость разработки отдельных подсистем проектируемой системы и параллельность выполнения работ. Следование волновой модели позволило осуществлять поэтапную разработку, а применение методов фазовых распределений и порядковых статистик дало возможность охватить оценками довольно широкий спектр проектов, а в некоторых случаях получить явные выражения для вероятностных распределений и моментов.

Заключение

Рассмотренный в работе подход пригоден не только для оценки временных характеристик процесса разработки ИУС в структуре SOS, но и для

оценки самой системы систем. Близкая к SOS теория активных систем развивается в Институте проблем управления РАН В. Н. Бурковым, Д. А. Новиковым и др. [13]. Применение технологии виртуального прототипирования и комбинированного моделирования ИУС, разрабатываемых в среде SOS, существенно ускоряет процесс и расширяет возможности обоснования принимаемых решений. Характерная черта SOS, связанная с относительной независимостью входящих в нее организаций, делает возможным более широкое практическое использование моделей параллельного выполнения работ, базирующихся на методах порядковых статистик. Причем, осуществляя аппроксимацию вероятностных распределений случайных величин, входящих в такие модели, и используя подходы марковизации [14], можно существенно упростить применяемый формальный аппарат и расширить возможности его использования для изучения сложных систем.

Список литературы

1. **System** of Systems Engineering. Innovation for 21st century / Ed. by M. Jamshidi. John Wiley & Sons, 2009. 609 p.
2. **Фатрелл Р. Т., Шафер Д. Ф., Шафер Л. И.** Управление программными проектами: достижение оптимального качества при минимуме затрат. М.: Вильямс, 2003. 1136 с.
3. **Acheson P., Pape L., Dagli C., Kilicay-Ergin, Colombi J., Harris H.** Understanding System of Systems Development Using an

Agent-Based Wave Model // Proc. of the Complex Adaptive Systems Conference, Washington D.C, November. 2012. P. 14–16.

4. **Литвин Ю. В., Ноздрин К. С., Полков А. А.** Система виртуального прототипирования и моделирования информационно-управляющих систем реального времени // Информационные технологии. 2017. Т. 23, № 5. С. 370–375.

5. **Balakrishnan N.** Permanents, Order Statistics, Outliers and Robustness // Rev. Mat. Complut., 2007. N 1. P. 7–107.

6. **Литвин Ю. В.** Порядковые статистики оценки времени параллельного выполнения работ проектов с ограниченными ресурсами // Проблемы экономики и менеджмента. 2013. № 11. С. 41–46.

7. **Литвин В. Г., Мусатов А. А., Литвин Ю. В.** Определение требований ко времени выполнения запросов компонентами системы реального времени методами бутстрепа // Автоматизация процессов управления. 2015. № 3. С. 41–48.

8. **Саати Т. Л.** Принятие решений при зависимостях и обратных связях. Аналитические сети. М.: Изд-во ЛКИ, 2008. 360 с.

9. **Бочаров П. П., Печинкин А. В.** Теория массового обслуживания. М.: Изд-во РУДН, 1995. 529 с.

10. **Бочаров П. П., Литвин В. Г.** Методы анализа и расчета систем массового обслуживания с распределениями фазового типа (обзор) // Автомат. и телемех. 1986. № 5. С. 5–23.

11. **Tijms H. C.** A First Course in Stochastic Models, John Wiley & Sons Ltd, 2003. 482 p.

12. **Литвин В. Г., Чернышова Е. В., Литвин Ю. В.** Оценка показателей работы вычислительной системы реального времени, содержащей узлы с параллельным выполнением процессов // Информационные технологии и вычислительные системы. 2014. № 2. С. 16–25.

13. **Бурков В. Н., Новиков Д. А.** Теория активных систем (история развития и современное состояние) // Проблемы управления. 2009. № 3.1. С. 29–35.

14. **Рыков В. В., Козырев Д. В.** Основы теории массового обслуживания. Основной курс: марковские модели и методы марковизации. М.: ИНФРА-М, 2016. 223 с.

V. G. Litvin, D. Sc., Professor, Chief Specialist, e-mail: litvg@mail.ru,
Joint Stock Company "Research Institute for Automated Apparatus
named after Academician V. S. Semenikhin" (JSC "NIIAA"), Moscow,
Yu. V. Litvin, PhD, Director of the Center, e-mail: litvinj@simplecs.ru,
Scientific-research Institute of Economics and management in gas industry
(LLC "NIIgazeconomika", JSC "Gazprom"), Moscow

About Assessment of Time Development Large-Scale Management Information System, Created in Structure System of Systems

New approach to assessment of time project development the information and management systems (IMS) in an organization structure system of systems (SOS) [1] according to wave model and technology of the virtual prototyping, conceptual and simulation — analytic models. SOS is used in case determination of cast on development different large-scale projects, in particular, projects of informatization administrative processes, energy projects and etc. It is caused by the fact that separate subsystems of IMS differ from each other with an essence processes of automation and require the appropriate competences their execution. Design process, as a rule, is multiphase with step-by-step with constant accumulation knowledge of future system and its surrounding, and step-by-step solution of problems. For the analysis of projects with structure SOS it is accepted wave model. Level of competences the respective area business and possibility use the accepted technology of development can be the main thing in case a choice performers. Key option of the technology considered in article is sharing of three directions of development and the analysis [4, 5]: 1) conceptual modeling; 2) virtual prototyping; 3) combined simulation and analytical modeling. As development system in the environment SOS is executed parallelly, an essential role is played at the same time by models assessment of time response characteristics parallelly developing processes. Application of the offered approaches and methods is shown by examples calculation probable characteristics runtime of projects.

Keywords: management information systems, wave model of project, system of systems, phase distributions

References

1. **System of Systems Engineering. Innovation for 21st century**, Ed. by M. Jamshidi, John Wiley & Sons, 2009, 609 p.
2. **Fatrel R. T., Shhafer D. F., Shafer L. I.** *Upravlenie programmnyimi proektami: dostizhenie optimal'nogo kachestva pri minimume zatrat*, Moscow, Vil'jams, 2003, 1136 p. (Safer D. F., Fatrel R. T., Safer L. I. Management program projects: achievement of optimum quality, Moscow, Williams, 2003. 1136 p.) (in Russian).
3. **Acheson P., Pape L., Dagli C., Kilicay-Ergin, Colombi J., Harris H.** Understanding System of Systems Development Using an Agent-Based Wave Model, *Proceedings of the Complex Adaptive Systems Conference, Washington D.C., November, 2012*, pp. 14–16.
4. **Litvin Ju. V., Nozdrin K. S., Polkov A. A.** Sistema virtual'nogo prototipirovaniya i modelirovaniya informacionno-upravljajushhih sistem real'nogo vremeni, *Informacionnye tehnologii*, 2017, vol. 23, no. 5, pp. 370–375. (Litvin Yu. V., Nozdrin K. S., Polkov A. A. System of virtual prototyping and simulation of management information real-time systems, *Information technologies*, 2017, vol. 23, no. 5, pp. 370–375) (in Russian).
5. **Balakrishnan N.** Permanents, Order Statistics, Outliers and Robustness, *Rev. Mat. Complut.*, 2007, no. 1, pp. 7–107.
6. **Litvin Ju. V.** Porjadkovye statistiki ocenki vremeni paralel'nogo vypolneniya rabot proektov s ogranichenennymi resursami, *Problemy jekonomiki i menedzhmenta*, 2013, no. 11, pp. 41–46 (Litvin Yu. V. Ordinal statistics of assessment time parallel execution operations of projects with limited resources, *Problems of economy and management*, 2013, no. 11, pp. 41–46) (in Russian).
7. **Litvin V. G., Musatov A. A., Litvin Ju. V.** Opredelenie trebovanij ko vremeni vypolnenija zaprosov komponentami sistemy real'nogo vremeni metodami butstrepa, *Avtomatizacija processov upravlenija*, 2015, no. 3, pp. 41–48. (Litvin V. G., Musatov A. A., Litvin Yu. V. Determination requirements time execution of requests components real-time system by methods of bootstrap // *Automation of management processes*, no. 3, 2015, pp. 41–48) (in Russian).
8. **Saati T. L.** Prinjatje reshenij pri zavisimostjah i obratnyh svjazjah. Analiticheskie seti, Moscow, Izd-vo LKI, 2008, 360 p. (Saati T. L. Decision-making in case of dependences and back couplings. Analytical networks, Moscow, LKI publishing house, 2008, 360 p.) (in Russian).
9. **Bocharov P. P., Pechinkin A. V.** Teorija massovogo obsluzhivaniya, Moscow, Izd-vo RUDN, 1995, 529 p. (Bocharov P. P., Pechinkin A. V. Queuing theory, Moscow, RUDN publishing house, 1995, 529 p.) (in Russian).
10. **Bocharov P. P., Litvin V. G.** Metody analiza i rascheta sistem massovogo obsluzhivaniya s raspredelenijami fazovogo tipa (obzor), *Avtomat. i telemeh.*, 1986, no. 5, pp. 5–23. (Bocharov P. P., Litvin V. G. Methods of analysis and calculation queuing systems with distributions of phase type // *Automation and Remote Control*, no. 5, 1986, pp. 5–23) (in Russian).
11. **Tijms H. C.** A First Course in Stochastic Models, USA, John Wiley & Sons Ltd, 2003, 482 p.
12. **Litvin V. G., Chernyshova E. V., Litvin Ju. V.** Ocenka pokazatelej raboty vychislitel'noj sistemy real'nogo vremeni, sodержashhej uzly s paralel'nym vypolneniem processov, *Informacionnye tehnologii i vychislitel'nye sistemy*, 2014, no. 2, pp. 16–25 (Litvin V. G., Chernyshova E. V., Litvin Yu. V. Assessment indices of operation computing system real time, containing nodes with parallel execution of processes, *Information technologies and computing systems*, 2014, no. 2, pp. 16–25) (in Russian).
13. **Burkov V. N., Novikov D. A.** Teorija aktivnyh sistem (istorija razvitija i sovremennoe sostojanie), *Probl. upravlj.*, 2009, no. 3.1, pp. 29–35. (Burkov V. N., Novikov D. A. Theory active systems (history development and current state), *Probl. Control*, 2009, no. 3.1, pp. 29–35) (in Russian).
14. **Rykov V. V., Kozyrev D. V.** *Osnovy teorii massovogo obsluzhivaniya. Osnovnoj kurs: markovskie modeli i metody markovizacii*, Moscow, INFRA-M, 20016. 223 p. (Rykov V. V., Kozyrev D. V. Bases of queuing theory. Basic course: Markov models and methods markovizatsii, Moscow, INFRA-M, 20016. 223 p.) (in Russian).

БЕЗОПАСНОСТЬ ИНФОРМАЦИИ CRYPTOSAFETY INFORMATION

УДК 004

Р. М. Алгулиев, акад. НАН Азербайджана, д-р техн. наук, проф., rasim@science.az,

Я. Н. Имамвердиев, канд. техн. наук, доц., e-mail: yadigar@lan.ab.az,

Л. В. Сухостат, канд. техн. наук, e-mail: lsuhostat@hotmail.com,

Институт информационных технологий национальной академии наук Азербайджана

Киберфизические системы: основные понятия и вопросы обеспечения безопасности

Данное исследование нацелено на выявление, классификацию и анализ существующих исследований по вопросам безопасности киберфизических систем, чтобы лучше понять, как безопасность на самом деле осуществляется при работе с киберфизическими системами. Рассматриваются оценка последствий кибератак, моделирование и обнаружение атак и разработка архитектуры безопасности. Описаны основные типы атак и классификация угроз на киберфизические системы. Показаны направления будущих исследований.

Ключевые слова: киберфизическая система, безопасность киберфизической системы, атаки на киберфизическую систему, угрозы безопасности киберфизической системы, обнаружение вторжений

Введение

Киберфизические системы (cyber-physical systems, CPS) в целом можно охарактеризовать как комплексные сетевые системы управления, которые сочетают в себе физические элементы реального мира с вычислительными элементами в киберпространстве [1, 2]. Они интегрируют физические процессы, распределенные датчики, актуаторы (исполнительные механизмы) и встраиваемые компьютеры через сеть связи и, как ожидается, должны превзойти традиционные встраиваемые системы в различных аспектах, таких как эффективность, безопасность, надежность, устойчивость и адаптивность [3].

Внедрение CPS связано с концепцией "Индустрия 4.0" [4], которая формирует процесс объединения технологий и знаний, обеспечивая автономность, надежность, системность, контроль без участия человека.

Ключевые технологические тенденции, лежащие в основе CPS, включают: "Интернет вещей", смарт-технологии, облачные вычисления и т. д.

Однако при увеличении взаимодействия между физическими и киберсистемами физические системы становятся все более восприимчивы к уязвимостям безопасности в киберсистеме. Убедиться в том, что система находится в безопасности во время взаимодействия с другой системой, является важным вопросом анализа CPS.

Несмотря на то, что значительный прогресс был достигнут при обеспечении надежной защиты от кибератак, безопасность CPS поднимает целый ряд

новых проблем. Проблемы по обеспечению безопасности CPS включают: моделирование угроз безопасности, разработки формального подхода к оценке уязвимостей CPS и проектирование надежных и отказоустойчивых архитектур для обработки быстро развивающихся кибер- и физических угроз.

Однако, традиционные подходы к безопасности информационных технологий не рассматривают взаимодействие между физическими устройствами и кибератаками. На самом деле, злоумышленник может атаковать не только киберсети, но и непосредственно физические элементы CPS.

Цель данной работы заключается в выявлении, классификации и анализе существующих исследований в области безопасности CPS для того, чтобы лучше понять, как на самом деле обеспечивается безопасность таких систем. Задачи данного исследования состоят в предоставлении картины состояния дел по безопасности CPS, помогая исследователям и практикам в поиске ограничений и недостатков современных исследований по архитектуре CPS и обнаружении вторжений.

В статье представлена модель работы CPS, приводится ее архитектура, а также угрозы безопасности таких систем, предлагается "дерево" атак на CPS, рассматриваются работы в области обеспечения информационной безопасности, которые предлагают различные стратегии и меры для поддержания желаемого уровня защиты CPS, представлены некоторые заключительные замечания и открытые вопросы в области безопасности.

О киберфизических системах

А. Что такое киберфизическая система?

Этот термин был предложен Хелен Джилл в 2006 г. на семинаре NSF CPS Workshop, проводимом Национальным научным фондом США. Сейчас CPS включены в приоритетные списки инноваций США и ряда европейских стран.

- С точки зрения информатики [5] CPS являются интеграцией вычислительных и физических процессов. Они включают встраиваемые компьютеры, сетевые мониторы и контроллеры, как правило, с обратной связью, где физические процессы влияют на вычисления, и наоборот.
- С точки зрения технологий автоматизации [6] CPS — это специализированные системы, деятельность которых контролируется и управляется с помощью вычислительных и коммуникационных ядер, встроенных в объекты и структуры физической среды.
- Согласно Национальному научному фонду США CPS будущего будут намного превышать существующие системы по производительности, адаптивности, отказоустойчивости, безопасности и удобству использования.

Б. Технические предпосылки создания киберфизических систем:

1) большое число используемых устройств на основе встроенных процессоров и, следовательно, увеличение памяти для хранения данных;

2) объединение различных технологических тенденций в большие системы: "Интернет вещей" и др.;

3) с ростом объемов информации необходимо передать часть управления CPS, сохраняя человека в контуре управления (human in the loop) [7].

В. Отличительные признаки киберфизических систем:

- взаимодействие систем через глобальную сеть Интернет;
- обеспечение надежности работы систем при централизованном автоматическом управлении;
- наличие единого киберпространства, которое обеспечивает обмен как внутри систем, так и с окружающей средой, а также защиту информации в виде криптосерверов, межсетевых экранов, антивирусов и т. д.;
- устойчивость систем, обеспечиваемая за счет автоматизированного интеллектуального управления.

Модель работы киберфизической системы

Текущее состояние CPS можно описать путем захвата значений важных переменных процесса. Два вида важных процессов или переменных состояния в CPS включают в себя: 1) измеренные переменные, представляющие данные, полученные с помощью сенсоров; 2) управляющие переменные, представляющие управляющие сигналы [8]. Нормальное значение некоторого параметра процесса

называется контрольной точкой (set point). В CPS расстояние между значениями переменных процесса и соответствующими контрольными точками рассчитывается контроллерами. После вычисления этого смещения контроллеры, используя сложный набор уравнений, разрабатывают стратегию и вычисляют новые переменные актуации или управления. Полученное управляемое значение посылается к соответствующему актуатору, чтобы держать процесс ближе к определенной контрольной точке [9].

Контроллеры также посылают полученные изменения на основные серверы управления и выполняют выбранные из них команды. В CPS операторы системы должны быть осведомлены о текущем состоянии контролируемых объектов. Таким образом, графический пользовательский интерфейс (graphical user interface, GUI), называемый интерфейсом "человек—машина" (human—machine interface, HMI), предоставляет текущее состояние контролируемого объекта человеку-оператору.

На рис. 1 представлена схема работы CPS.

В общем процесс работы CPS можно разделить на четыре основных этапа:

Мониторинг. Мониторинг физических процессов и окружающей среды является основной функцией CPS. Он также используется для обратной связи с любыми последними действиями, предпринятыми CPS, и проверяет правильность операций, которые будут происходить в будущем.

Работа в сети (networking). Этот этап заключается в агрегации (диффузии) данных. В то же время различные приложения должны взаимодействовать с сетевой коммуникацией.

Вычислительная обработка (computing). Этот этап для осмысления и анализа данных, собранных в ходе мониторинга, чтобы проверить, удовлетворяет ли физический процесс заранее определенным критериям.

Приведение в действие (actuation). Этот этап предполагает действия, определенные во время фазы вычислений. Приведение в действие может осуществлять различные события, такие как исправление киберповедения CPS, изменение физического процесса.

Архитектура CPS часто состоит из двух основных слоев [10, 11]:

1) киберслоя, который состоит из корпоративной сети, сети управления и демилитаризованной зоны (ДМЗ);

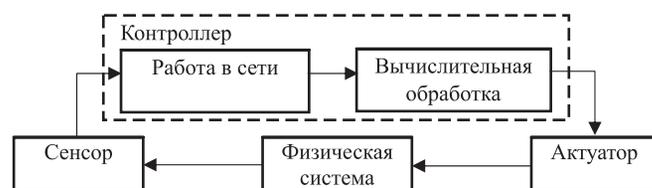


Рис. 1. Упрощенная структурная схема киберфизической системы

2) физического слоя, который состоит из сенсоров, исполнительных устройств (актуаторов) и физических устройств.

Корпоративная сеть, содержащая рабочие станции, сотрудников и сервер приложений, отвечает за управление бизнесом и взаимодействие с клиентами. Сеть управления состоит из контроллеров, таких как PLC, сервера управления и HMI. ДМЗ определяется как отдельный сегмент сети, который подключается напрямую к брандмауэрам [12, 13]. Серверы, включая сервер данных, хранилище данных и веб-серверы (для взаимодействия корпоративных клиентов), содержащие данные из CPS, которым нужно получить доступ из корпоративной сети, переносятся в этот отдельный сегмент для повышения уровня безопасности [13]. Киберслой часто использует промышленные протоколы, такие как DNP3 [14], 61 850 [15] и Modbus [16], чтобы обмениваться данными с устройствами физического уровня.

Новая вычислительная парадигма, известная как кибер-физико-социальные вычисления или физико-кибер-социальные вычисления [17], возникла из CPS и кибер-социальных систем (cyber-social systems, CSS). К сожалению, конструкция кибер-физико-социальных систем (cyber-physical-social systems, CPSS) является сложной задачей [18]. Сложно равномерно смоделировать кибер, физическое и социальное пространства. Таким образом, создание эффективного, универсального, безопасного дизайна для CPSS является сложной задачей. CPSS расширяют CPS и включают социальное пространство, признаки участия и взаимодействия людей [19].

Архитектура киберфизической системы

Модель работы CPS можно рассмотреть по различным уровням детализации. Наиболее общая архитектура CPS разбивается на пять фундаментальных уровней [20, 21]. На рис. 2 представлена вертикальная структура CPS от физического уровня до уровня приложений.

Физический/МАС-уровень. Физический уровень закладывает основу CPS-архитектуры. Media access control (MAC) — это уровень управления доступом к среде, относящийся ко второму уровню модели ISO/OSI (сетевому уровню). Физический уровень состоит из сенсоров, актуаторов, которые соединены друг с другом по беспроводной или проводной сети. Примерами являются 2G/3G/4G, WiFi, ZigBee, Bluetooth, WiMAX, RFID считыватели и метки и проводные технологии (PLC, NC и др.). 6LoWPAN

(IPv6 over Low power Wireless Personal Area Networks) является протоколом сетевого уровня и может использоваться с любым физическим и канальным уровнем. Используется для соединения ZigBee, Bluetooth и других систем с Интернетом (выполняет роль маршрутизатора).

Сетевой уровень (Network Layer). На этом уровне происходит маршрутизация пакетов на основе преобразования MAC-адресов в сетевые. Он использует протоколы IPv4/IPv6 и RPL ("Ripple" routing protocol).

Транспортный уровень (Transport Layer). На данном уровне пакеты разбиваются на малые фрагменты. Наиболее распространенные протоколы транспортного уровня включают: TCP, UDP и ICMP.

Промежуточный уровень (Middleware Layer). Он выполняет операции по управлению терминалом, преобразованию протоколов, осуществляет продвижение данных и аутентификацию для CPS-приложений. На этом уровне архитектура ETSI M2M обеспечивает взаимодействие транспортного уровня с уровнем приложений. Промежуточный слой состоит из модуля адресации и профилирования, модулей поиска и выбора устройства, модуля управления и т. д.

Уровень приложений (Application Layer). Этот уровень хранит, анализирует и обновляет информацию, полученную от предыдущих уровней. Он принимает решения управления, которые могут быть визуализированы с помощью интерфейса виртуального прототипа (virtual prototype interface).

Есть несколько принципов проектирования безопасности, которые могут быть полезны при по-

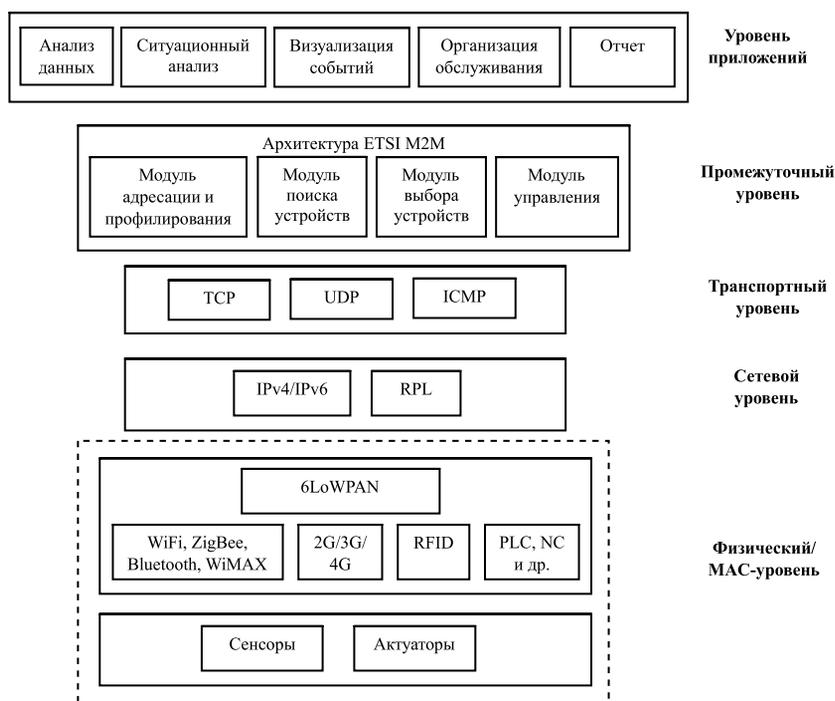


Рис. 2. Общая архитектура киберфизической системы

строении систем управления, которые могут выжить при атаках [22–24]: избыточность (*redundancy*), несогласованность (*diversity*), принцип наименьших привилегий (*principle of least-privilege*) и разделение привилегий (*separation of privilege*).

Угрозы безопасности киберфизических систем

Киберугрозы влияют на: *конфиденциальность*, которая необходима для поддержания безопасности личных данных пользователей в CPS и предотвращения попытки злоумышленника изменить состояние физической системы путем "подслушивания" каналов связи между датчиками и контроллером, а также между контроллером и актуатором; *целостность*, когда данные или ресурсы могут быть изменены без разрешения; *доступность*, когда происходят сбои в вычислительной технике, управлении, коммуникации, оборудовании; *достоверность*, когда необходимо подтвердить, что обе участвующие стороны есть действительно те, за кого они себя выдают [25, 26].

Одной из основных характеристик киберугроз является то, что они масштабируемы, т. е. они легко автоматизируются и тиражируются, и следует ожидать, что они распространяются свободно через ненадежные домены. Киберфизические угрозы представляют собой угрозы, которые берут начало в киберпространстве, но оказывают влияние на физическое пространство системы.

Классификация угроз CPS включает [27]: подмену объектов (*Spoofing identity*), модификацию данных (*Tampering with data*), отказ от авторства (*Repudiation of origin*), разглашение информации (*Information disclosure*), повышение привилегий (*Elevation of privilege*), отказ в обслуживании (*Denial of service, DoS*).

В [28] выделены ключевые проблемы для обеспечения безопасности CPS:

- понимание угроз и возможных последствий атак;
- определение уникальных свойств CPS и их отличия от безопасности традиционных информационных технологий;
- обсуждение механизмов безопасности, применимых к CPS.

Вместе с тем для понимания новых классов угроз на CPS, например, на смарт-сети и SCADA (Supervisory for Control And Data Acquisition) системы, полезно характеризовать взаимодействия между областью, которая является источником угрозы, и областью, которая подверглась ее влиянию [29].

"Дерево" атак на киберфизические системы

По результатам анализа существующих исследований по вопросам безопасности на рис. 3 предлагается "дерево" атак на основе функциональной модели CPS (см. рис. 1):

- а) атаки на сенсорные устройства (*Sensing*);

- б) атаки на актуаторы (*Actuation*);
- в) атаки на вычислительные компоненты (*Computing*);
- г) атаки на линии связи/коммуникации (*Communication*);
- д) атаки на обратную связь (*Feedback*).

а. Исследователи выявили угрозы и уязвимости, влияющие на сенсоры CPS (такие как Injecting false radar signals, Dazzling cameras with light, GPS Spoofing и др.) [30]. Были выделены два основных типа атак: DoS- и спуфинг-атаки. Так как CPS тесно связаны с физическим процессом, в который они встроены, достоверность и точность процесса получения данных должны быть обеспечены. Безопасность сенсоров нуждается в методах для стимулирования физической аутентификации, так что любым данным, полученным от физического процесса, можно доверять [31].

б. Djouadi и соавторы [32] проанализировали влияние кибератак на актуаторы и рассмотрели два класса, охватывающих широкий спектр потенциальных атак: атака конечной энергии (*Finite Energy Attack*), которая включает, например, потерю и модификацию персональных пакетов, атаку конечного времени (*Finite Time Attack*), импульсные атаки и ограниченную атаку (*Bounded Attack*), которая приводит к подавлению управляющего сигнала.

Безопасность управления актуацией относится к тому, что во время пассивно-активного или ак-

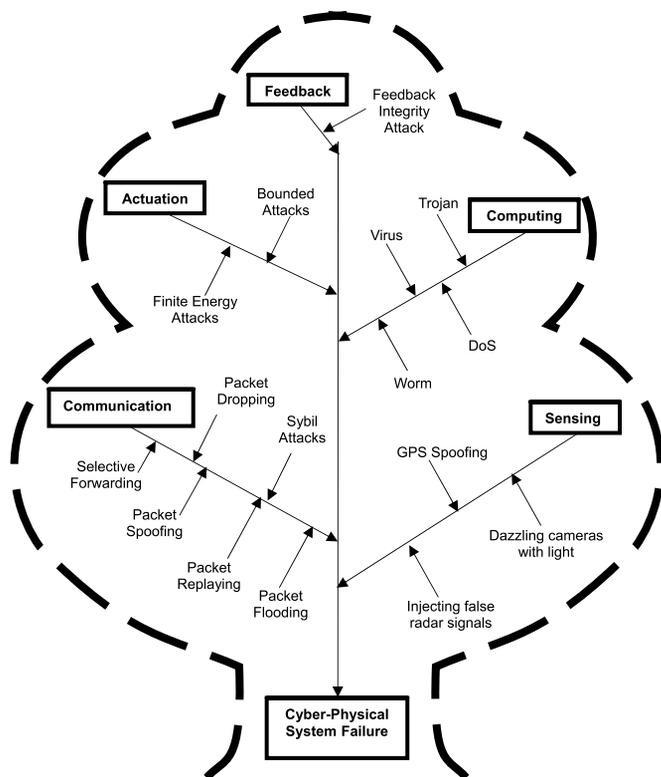


Рис. 3. Древоподобная диаграмма атак на киберфизические системы

тивного режима работы ни одно действие не может происходить без соответствующего разрешения. Спецификация разрешений должна быть динамичной, так как требования к CPS меняются со временем.

в. Атаки на вычислительные ресурсы были подробно рассмотрены в работе [33] и включают трояны (*Trojans*), вирусы (*Viruses*), черви (*Worms*) и *DoS*-атаки, где предоставляется информация о том, какие методы интеллектуального анализа данных могут быть использованы для повышения кибербезопасности.

Вредоносная атака может скрытно повредить CPS. Так как существуют нарушения и ошибки измерений в системах управления, механизмы обнаружения должны гарантировать, что эти нормальные ошибки не будут вызывать ложную тревогу. Это дает атакующему пространство, чтобы скрыться. С точки зрения атакующего, построение стратегии ложных атак, как правило, сталкивается с целым рядом факторов, ресурсов и ограничений безопасности.

г. Коммуникационные атаки включают избирательную пересылку (*Selective Forwarding*), выбрасывание пакетов (*Packet Dropping*), спуфинг (*Packet Spoofing*), лавинную рассылку пакетов (*packet flooding*), повторное воспроизведение пакетов (*Packet Replaying*) и *Sybil*-атаки (могут быть использованы в целях нарушения распределения ресурсов между узлами в пользу вредоносного), нарушающие маршрутизацию пакетов системы [34].

Важным аспектом CPS является то, что они объединены в сеть по своей природе. Это не только позволяет им сформировать сеть для объединения данных, но также доставку окончательных объектов и принятие согласованных мер реагирования (как в пассивно-активном, так и активном режимах работы). Безопасность связи нуждается в разработке протоколов для обеспечения как между, так и внутри CPS-связей от активных (источников помех) и пассивных (перехватчиков).

д. В работе [10] была предложена трехслойная логическая модель CPS и метамодель кибератак, где система подвергается атаке на целостность обратной связи (*Feedback Integrity Attack*), при этом только часть управляющих сигналов сохраняет свою целостность.

Безопасность обратной связи (*Feedback Security*) относится к тому, что системы управления в CPS, которые обеспечивают необходимую обратную связь для актуации, защищены. Современные решения в области безопасности сосредоточены только на безопасности данных, но их влияние на алгоритмы оценки и управления должны быть изучены для обеспечения углубленной защиты CPS [35].

Любое вмешательство в данные может привести к ошибкам в будущих требованиях к их обработке. Если злоумышленник может только захватить и

переслать реальные пакеты данных, то эффективный метод атаки — записать некоторые нормальные данные и воспроизвести их, чтобы избежать обнаружения.

Это называется атакой методом записи и повторной передачи блоков зашифрованного теста (*Replay Attack*). Если злоумышленник изменяет реальные данные путем получения ключа для защищенной связи (*Communication Key*) или захватом некоторых сетевых устройств, это называется атакой на целостность данных (*Integrity Attack*). Безопасность хранения включает в себя разработку решений для обеспечения безопасности хранимых данных в CPS-платформах от физического или кибервзлома.

Основные направления исследований

Анализ современных публикаций по данной теме показал, что киберфизические системы — одна из самых актуальных и перспективных технологий [25, 36]. Ученые из различных стран (Германия, Китай, США и др.) посвятили данной технологии тысячи публикаций, которые исследуют создание таких систем [37].

Исследования по безопасности CPS имеют следующие направления:

1. *Оценка последствий кибератак.* Сложные и изощренные атаки призваны нанести значительный ущерб кибер- и физическим характеристикам CPS. Например, Stuxnet [38, 39] является первым вредоносным программным обеспечением, специально разработанным для нанесения физического ущерба производственным инфраструктурам (перепрограммирование систем управления путем модификации кода PLC). Таким образом, необходимо оценить воздействие кибератак на нормальное функционирование физических процессов. В таких ситуациях крайне важно не только продемонстрировать и оценить разрушительное воздействие кибератак, но и дать количественную оценку последствий и в конечном итоге обеспечить необходимость наличия конкретных киберактивов.

2. *Моделирование уязвимостей и атак на CPS.* Модели атак и уязвимостей используются для выявления слабых сторон в CPS-системах для поддержания стратегии их поиска и содействия пониманию атак. Для обеспечения безопасности CPS необходима разработка моделей атак в целях их оценки и принятия адекватных мер противодействия.

3. *Обнаружение атак на CPS.* Разработка алгоритмов обнаружения и контрмер для всех известных атак заранее необходима для снижения воздействия атак в ограниченное время и сведения к минимуму повреждений системы.

4. *Разработка архитектуры безопасности.* Развитие киберфизических систем сдерживается факторами безопасности. Основная задача, стоящая при проектировании сложных архитектур CPS, со-

стоит в тестировании и валидации "безопасного проектирования" для обеспечения безопасности и надежности физических и киберкомпонент.

Необходимо разработать новые надежные алгоритмы управления и оценивания, которые рассматривают более реалистичные модели атак с точки зрения безопасности. Исследованием данной проблемы занимаются ведущие университеты ряда технологически развитых стран. Рассмотрим подробнее эти направления.

1. Оценка последствий атак на киберфизические системы

Оценка последствий кибератак на CPS может быть успешно реализована в соответствии с моделью оценки риска и алгоритмом, предложенным в работе [40], который рассматривает три показателя: тяжесть атаки, вероятность успеха атаки и следствие атаки. Кривая изменения риска может помочь пользователям лучше понять системный риск в режиме реального времени и отреагировать на него своевременно. Кроме того, она также может быть использована для предсказания риска в будущем.

В [41] предложен метод оценки последствий распространения атак в киберфизических системах, оценивающий прямые и косвенные последствия нападений на параметры управления, включая измерения CPS-сенсоров и сигналов контроллера. Сравниваются нормальное поведение системы без каких-либо сбоев и аномальное поведение при атаках. Параметры системы делятся на два класса причинно-следственных параметров, которые могут быть одинаковыми или могут отличаться друг от друга. Наконец, предложены новые показатели, которые могут быть использованы для количественной оценки уровня важности каждого параметра в физическом процессе. Используя полученные количественные значения, в данной работе определяют приоритеты в показаниях сенсоров и сигналов управления в зависимости от их чувствительности к проводимым атакам. В отличие от большинства предлагаемых методов, применимых к атакам, которые заставляют физический процесс выключаться (например, [42, 43]), предложенный метод может рассматривать атаки, которые не обязательно приводят к отключениям системы.

Genge и соавторы [44] описывают проблему, касающуюся того, как сетевые параметры (потеря пакетов, задержка связи, логика управления временем планирования и сетевой трафик) могут повлиять на последствия атак. Основной вклад этой статьи состоит в том, что были определены наиболее важные параметры, которые могут повлиять на устойчивость физических процессов. Авторы отметили, что параметры связи (например, задержка связи) имеют ограниченное воздействие на результат проведенных атак и параметры планирования

задач могут повлиять на устойчивость физических процессов.

В [45] был предложен метод для оценки последствий кибератак на физические процессы. Исследование основывается на поведенческой оценке физических процессов и анализе чувствительности. Метод оценки вычисляет ковариацию наблюдаемых переменных до и после выполнения индивидуальных атак против управляющих переменных. Одной из основных особенностей этого метода является его применимость к ситуациям, когда физический процесс недоступен. Этот метод рассматривает только отдельные (индивидуальные) нападения на управляющие сигналы.

Язык для описания возможных атак на CPS и их последствия был предложен в [46]. Основным достоинством языка является определение и описание признаков, выражающих атаки и контрмеры против CPS. Несмотря на то, что они не рассматриваются в процессе оценки безопасности, авторы считают, что предлагаемый язык описания атак может быть использован для оценки уровня безопасности.

В [47] было предложено использовать граф атак и метод мультикритериального принятия решений (Multiple Criteria Decision Making, MCDM) для оценки безопасности процесса управления мощностью.

Wei и соавторы [48] обсудили потенциальные кибератаки, их последствия и защитные требования к энергосетям. Кроме того, был предоставлен обзор проблем безопасности и стратегии в энергосетях.

Подход на основе теории игр к оценке безопасности смарт-сетей был предложен в работе [49]. В первую очередь авторы сосредоточились на киберфизической безопасности обширной территории мониторинга, защите и контроле с точки зрения скоординированных кибератак.

Varshney и соавторы [50] предложили использовать параллельные вычисления для быстрого и точного анализа интегрированной безопасности энергосистем. Предлагаемый подход дает возможность исследовать взаимное влияние рассмотренных параметров управления и определяет наиболее значимые из них. Эта информация дает хорошее представление о наиболее уязвимых контурах управления в CPS.

2. Моделирование уязвимостей и атак на киберфизические системы

В работе [51] модель атаки записи (*Replay Attack*) на CPS была определена и проанализирована эффективность системы управления. Были описаны отношения между потерей производительности, скоростью обнаружения и силой сигнала аутентификации. Также была представлена методика оптимизации зашумленных сигналов аутентификации на основе компромисса между желаемой эф-

фективностью обнаружения и допустимой потерей производительности управления. В статье было предложено введение сигнала аутентификации в систему через случайные интервалы времени, а не непрерывно, таким образом влияя на производительность в течение некоторого времени.

На основе инструментов из теории структурированных систем в работе [52] описываются неподдающиеся обнаружению атаки, а также характеризуется вычислительная сложность задачи их идентификации.

В статье [53] Gianì и соавторы представили и охарактеризовали неочевидные кибератаки и предложили контрмеры против них с использованием заведомо безопасных устройств синхронизированных векторных измерений (Phasor Measurement Unit, PMU). Было показано, что $(p + 1)$ PMU устройств достаточно эффективны при отключении p атак. Более глубокая проблема с исследованием кибербезопасности компонентов SCADA/EMS энергосистемы связана с Grid-операциями. Атака имеет последствия только тогда, когда оператор сети введен в заблуждение, что приводит к компрометации данных. Поэтому полный и реалистичный анализ угроз кибербезопасности электрических сетей должен включать в себя как обычный технологический режим, так и чрезвычайные ситуации.

В работе [54] предлагается аспектно-ориентированное моделирование (АОМ) как мощный подход к моделированию для оценки безопасности CPS при использовании особых моделей атак. Оценка поведения системы в случае атак не должна требовать серьезных изменений в модели системы. В результате могут быть разработаны различные модели атак и на их основе создана библиотека.

Исследование атак, проблем оценки рисков, механизмов обнаружения и реагирования против систем управления было проведено в работе [55]. Авторы моделируют surge-атаки, bias-атаки и геометрические атаки в системах управления процессами.

В [56] был предложен подход к моделированию и оценке атак и защитных действий для CPS. Модель основана на стохастических сетях Петри (Stochastic Petri Nets, SPNs). В своем подходе были рассмотрены потери (attrition), неисправности (failure) и exfiltration failures. Определение оптимальной модели условий в CPS, таких как интервал обнаружения вторжений и моделирование уровня избыточности, является результатом исследования.

Авторы в [57] обратились к моделированию атаки, используя уязвимость информации, коммуникации и электрической сети, проанализировали уязвимости электрической сети с неполной информацией при использовании подхода из теории графов. Кроме того, был введен комплексный индекс киберфизической уязвимости и использование его для моделирования в режиме реального времени при демонстрации влияния атаки "Auroga".

3. Обнаружение атак на киберфизические системы

Определение оптимального времени для проведения атак на CPS было исследовано в работе [42]. В первую очередь авторы сосредоточились на воздействии DoS-атак на физические процессы, когда они выполняются в нужное время. Эксперименты показывают, что злоумышленники должны проводить атаки в определенный момент времени в целях нарушения физического процесса. В противном случае атаки будут иметь ограниченные последствия для процесса.

В работе [58] был предложен подход для обнаружения аномалий, которые являются результатом воздействия APT (*Advanced Persistent Threat*) угроз (например, прямой доступ к серверам баз данных, копирование больших объемов данных). Обнаружение аномалий в данном подходе возможно только благодаря использованию комбинации различных правил, описывающих данную модель.

Предложенный в [59] подход объединил в себе ключевые принципы современных методов обнаружения троянов, влияющих на физические изменения изготавливаемых деталей в производстве.

Атаки внедрения ложных данных (*false data injection attacks*) были исследованы в работе [60]. Разработаны контрмеры повышения устойчивости методов фильтрации Калмана. Методика позволяет обнаруживать атаки быстро и точно [60].

В работе [61] представлена модель, охватывающая новый набор функций и настраиваемых алгоритмов распознавания образов для идентификации атак на целостность данных, влияющих на CPS. Эта информация может иметь большое значение для оператора при принятии обоснованных решений относительно согласованных действий и дальнейшего использования инфраструктуры. Предлагаемые системы обнаружения атак на целостность (Integrity Attack Diagnosis Systems, IADS) используют преимущества от слияния характеристик, принадлежащих двум различным представлениям сигналов, т. е. частоте и вейвлету. Они используются для обучения случайного "леса" на выявление атак целостности с учетом новых входящих потоков данных. Кроме того, предлагаемый способ обнаруживает неизвестные ранее данные, снижая потенциал неправильной классификации.

4. Разработка архитектуры безопасности

Авторы [52] описали единую структуру модели CPS и определили понятия выявляемости и идентифицируемости атаки по ее влиянию на выходные измерения.

Определение потенциальных уязвимостей важно в ходе систематической фазы проектирования в целях минимизации общих затрат, связанных с предоставлением и поддержанием их безопасности и надежности [62]. В данной работе рассматрива-

ются эти проблемы и предлагается инструмент, который позволяет во время фазы проектирования CPS выполнять систематический анализ моделирования угроз с использованием реальной системы мониторинга температуры железной дороги как частного случая.

В [37] был предложен гибридный подход для создания архитектуры надежных и отказоустойчивых систем. Sanislav и Micle [63] представили процедуру от "анализа к построению" для управляемой событиями мультиагентной модели развития для CPS с надежными характеристиками. Одно из реальных ограничений состоит в том, что многие CPS, такие как энергетические сети, имеют обширную, уже построенную инфраструктуру, и эти устаревшие системы часто слишком сложно и дорого заменить. Кроме того, рабочие среды CPS часто непредсказуемы, и моделировать все виды аномальных условий практически невозможно в тестовой среде. Повышение надежности работы этих систем влечет за собой работу с программным обеспечением, аппаратными средствами и физическими устройствами, которые были уже применены.

В целях достижения надежности CPS через построение процесса разработки Wang и соавторы [64] предложили надежный рабочий процесс совместимости CPS-услуг и времени/пространства выполнения. Однако такой подход на основе процесса работы также не имеет отношение к реальным CPS.

В работе [65] авторы обсудили концепцию и стратегии для создания надежной и отказоустойчивой CPS. Они определили отказоустойчивость как 3S-ориентированную модель (Stability, Security и Systematicnes), т. е. стабильность, безопасность и систематичность, а также указали проблемы, связанные с моделированием CPS.

Yoo и соавторы [66] обсудили уязвимости, требования к безопасности и архитектуру CPS.

Открытые вопросы

В данном разделе будут рассмотрены такие вопросы, как разработка методов повышения безопасности CPS, оценка последствий атак на такие системы, а также контрмеры против нарушения конфиденциальности и целостности данных в киберсистемах.

- *Разработка, применение и оценка последствий атак на CPS на основе методов машинного обучения.* В силу того, что существующее программное обеспечение для борьбы с кибератаками не способно справиться с передовыми атаками, необходимо разработать комбинированные подходы на основе машинного обучения и анализа угроз.
- *Разработка контрмер против нарушения конфиденциальности и целостности данных в CPS.* Это связано с быстро растущей проблемой DDoS-атак на системы.

- *Разработка методов верификации, оценки и тестирования механизмов безопасности в контексте надежности CPS.* Необходимо повысить внимание к аутентификации как наиболее главной форме защиты учетных записей и доступа к системе.
- *Разработка архитектуры для обеспечения надежности и устойчивости CPS.* Система должна поддерживать необходимую функциональность, несмотря на частичные отказы, и противостоять вредоносным атакам как на кибер-, так и физические компоненты.
- *Применение методов классификации для обнаружения аномалий в CPS.* Отклонение от нормального поведения при обнаружении вторжений показывает наличие аномалий в системе.

Заключение

CPS являются перспективной парадигмой для разработки текущих и будущих инженерных систем и, как ожидается, окажут важное влияние на реальный мир. Идея CPS акцентирует внимание на проектировании комплексных систем, а не киберили физической системы отдельно. Для того чтобы пролить свет на сегодняшние проблемы безопасности CPS, в статье был представлен обзор соответствующей литературы по обсуждению практических приложений и доминирующих областей исследований. Были исследованы проблемы атак в киберфизическом пространстве, которые имеют различные последствия и цели: изменить некоторые атрибуты безопасности, вызвать катастрофические повреждения системного оборудования и ресурсов, привести к производственным убыткам, поставить под угрозу жизнь и безопасность людей, вызвать нанесение ущерба окружающей среде.

CPS несут высокий потенциал для создания новых рынков и решений социальных рисков, но налагают высокие требования на качество, защиту, безопасность и неприкосновенность частной жизни [67–70]. основополагающие научные исследования необходимы для достижения предсказуемого уровня качества проверки и измерения, эффективной борьбы с внешними и внутренними изменениями, поддержки необходимых переходов между механической, электрической и программной инженериями, а также интеграции аспектов управления, проектирования и применения.

Список литературы

1. **Wu G., Sun J., Chen J.** A survey on the security of cyber-physical systems // Control Theory and Technology. 2016. Vol. 14, N. 1. P. 2–10.
2. **NIST**, Cyber-Physical Systems. URL: <https://www.nist.gov/el/cyber-physical-systems> (дата обращения: 27.01.17).
3. **Rawat D. B., Rodrigues J., Stojmenovic I.** Cyber Physical Systems: From Theory to Practice. USA: CRC Press, 2015. 570 p.

4. **Jazdi N.** Cyber physical systems in the context of industry 4.0 // Proceedings of the IEEE International Conference on Automation, Quality and Testing, Robotics. 2014. P. 1–4.
5. **Lee E. A.** Cyber physical systems: Design challenges // Proceedings of the 11th International Symposium on Object/Component/Service-Oriented Real-Time Distributed Computing. 2008. P. 363–369.
6. **Johansson K. H.** Control of Cyber-physical Systems: Fundamental Challenges and Applications to Transportation Networks // Proceedings of the 27th International Conference on Architecture of Computing Systems. 2014. XI p.
7. **Stankovic J. A.** Research directions for the Internet of Things // IEEE Internet of Things Journal. 2014. Vol. 1, N. 1. P. 3–9.
8. **Krotofil M., Cardenas A.** Resilience of process control systems to cyberphysical attacks // Proceedings of the 18th Nordic Conference on Secure IT Systems. 2013. P. 166–182.
9. **Kopetz H.** Real-Time Systems: Design Principles for Distributed Embedded Applications. USA: Springer, 2011. 378 p.
10. **Hahn A., Thomas R. K., Lozano I., Cardenas A.** A multi-layered and kill-chain based security analysis framework for cyber-physical systems // International Journal of Critical Infrastructure Protection. 2015. Vol. 11. P. 39–50.
11. **Krotofil M., Larsen J.** Are you threatening my hazards? // Proceeding of the 9th International Workshop on Security. 2014. P. 17–32.
12. **Hills R.** Common VPN security flaws, White Paper, NTA Monitor. 2005. URL: <http://www.nta-monitor.com/> (дата обращения: 25.01.17)
13. **Stouffer K., Falco J., Scarfone K.** Guide to Industrial Control Systems (ICS) Security // NIST Special Publication. 2011. P. 800–882.
14. **Majdalawieh M.** Security Framework for DNP3 and SCADA. Saarbruken, Germany: VDM Verlag. 2008. 160 p.
15. **Ozansoy C. R., Zayegh A., Kalam A.** Time synchronisation in a IEC 61850 based substation automation system // Proceedings of the 2008 Australasian Universities Power Engineering Conference. 2008. P. 1–7.
16. **Modbus-IDA**, Modbus Application Protocol Specification V.1.1b. URL: www.modbus.org/docs/Modbus_Application_Protocol_V1_1b.pdf. (дата обращения: 19.01.17)
17. **Sheth A., Anantharam P., Henson C.** Physical-cyber-social computing: an early 21st century approach // IEEE Intelligent Systems. 2013. Vol. 28, N. 1. P. 78–82.
18. **Wang F.-Y.** The Emergence of Intelligent Enterprises: from CPS to CPSS // IEEE Intelligent Systems. 2010. Vol. 25, N. 4. P. 85–88.
19. **Zeng J., Yang L. T., Lin M., Ning H., Ma J.** A survey: Cyber-physical-social systems and their system-level design methodology // Future Generation Computer Systems. 2016.
20. **Li H., Lai L., Poor H. V.** Multicast routing for decentralized control of cyber physical systems with an application in smart grid // IEEE Journal on Selected Areas in Communications. 2012. Vol. 30. P. 1097–1107.
21. **Koubaa A., Andersson B.** A Vision of Cyber-Physical Internet // Proceedings of the 8th International Workshop on Real-Time Networks. 2009. P. 1–6.
22. **Cardenas A. A., Amin S., Sinopoli B., Giani A., Perrig A., Sastry S.** Challenges for securing cyber physical systems // Workshop on Future Directions in Cyber-physical Systems Security. 2009. P. 1–4.
23. **Saltzer J. H., Schroeder M. D.** The protection of information in computer systems // Proceedings of the IEEE. 1975. Vol. 63, N. 9. P. 1278–1308.
24. **Avizienis A., Laprie J.-C., Randell B., Landwehr C.** Basic concepts and taxonomy of dependable and secure computing // IEEE Transactions on Dependable and Secure Computing. 2004. Vol. 1, N. 1. P. 11–32.
25. **Wang E. K., Ye Y., Xu X., Yiu S. M., Hui L. C. K., Chow K. P.** Security Issues and Challenges for Cyber Physical System // IEEE/ACM International Conference on on Cyber, Physical and Social Computing. 2010. P. 733–738.
26. **Sicari S., Rizzardi A., Grieco L. A., Coen-Porisini A.** Security, privacy and trust in Internet of Things: The road ahead // Computer Networks. 2015. Vol. 76. P. 146–164.
27. **Xu D., Tu M., Sanford M., Thomas L., Woodraska D., Xu W.** Automated security test generation with formal threat models // IEEE Transactions on Dependable and Secure Computing. 2012. Vol. 9, N. 4. P. 525–539.
28. **Xinlan Z., Zhifang H., Guangfu W., Xin Z.** Information Security Risk Assessment Methodology Research: Group Decision Making and Analytic Hierararchy Process // Second WRI World Congress on Software Engineering. 2010. P. 157–160.
29. **Neuman C., Tan K.** Mediating Cyber and Physical Threat Propagation in Secure Smart Grid Architectures // Second International Conference on Smart Grid Communications. 2011. P. 238–243.
30. **Brooks Z.** Hacking Driverless Vehicles // DEFCON. URL: <https://www.defcon.org/images/defcon-21/dc-21-presentations/Zoz/DEFCON-21-Zoz-Hacking-Driverless-Vehicles.pdf> (дата обращения: 24.12.16)
31. **Krotofil M., Larsen J., Gollmann D.** The process matters: Ensuring data veracity in cyber-physical systems // Proceedings of the 10th ACM Symposium on Information, Computer and Communications Security. 2015. P. 133–144.
32. **Djouadi S. M., Melin A. M., Ferragut E. M., Laska J. A., Dong J.** Finite Energy and Bounded Actuator Attacks on Cyber-Physical Systems // Proceedings of the IEEE European Control Conference. 2015. P. 3659–3664.
33. **Singhal A.** Data Warehousing and Data Mining Techniques for Cyber Security, USA: Springer Science + Business Media. 2007. 159 p.
34. **Mitchell R., Chen I. R.** Effect of intrusion detection and response on reliability of cyber physical systems // IEEE Transactions on Reliability. 2013. Vol. 62, N. 1. P. 199–210.
35. **Cardenas A. A., Amin S., Sastry S.** Research challenges for the security of control systems // Proceedings of the 3rd conference on Hot Topics in Security. 2008. P. 1–6.
36. **K. Wan, Man K. L., Hughes D.** Specification, Analyzing Challenges and Approaches for Cyber-Physical Systems (CPS) // Engineering Letters. 2010. Vol. 18, N. 3. P. 308–315.
37. **Sha L., Meseguer J.** Design of complex cyber physical systems with formalized architectural patterns // Software-Intensive Systems and New Computing Paradigms. Berlin: Springer-Verlag. 2008. P. 92–100.
38. **Karnouskos S.** Stuxnet worm impact on industrial cyber-physical system security // Proceedings of the 37th Annual Conference of the IEEE Industrial Electronics Society. 2011. P. 4490–4494.
39. **Collins S., McCombie S.** Stuxnet: the emergence of a new cyber weapon and its implications // Journal of Policing, Intelligence and Counter Terrorism. 2012. Vol. 7, N. 1. P. 80–91.
40. **Wu W., Kang R., Li Z.** Risk Assessment Method for Cyber Security of Cyber Physical Systems // Proceedings of 2015 the 1st International Conference On Reliability Systems Engineering. 2015. P. 523–527.
41. **Orojloo H., Abdollahi Azgomi M.** A method for evaluating the consequence propagation of security attacks in cyber-physical systems // Future Generation Computer Systems. 2017. Vol. 67. P. 57–71.
42. **Krotofil M., Cardenas A. A., Larsen J., Gollmann D.** Vulnerabilities of cyber-physical systems to stale data-determining the optimal time to launch attacks // International Journal of Critical Infrastructure Protection. 2014. Vol. 7. P. 213–232.
43. **Huang Y. L., Cardenas A. A., Amin S., Lin Z. S., Tsai H. Y., Sastry S.** Understanding the physical and economic consequences of attacks on control systems // International Journal of Critical Infrastructure Protection. 2009. Vol. 2. P. 73–83.
44. **Genge B., Siaterlis C., Hohenadel M.** Impact of network infrastructure parameters to the effectiveness of cyber attacks against industrial control systems // International Journal of Computers, Communications & Control. 2014. Vol. 7. P. 674–687.
45. **Genge B., Kiss I., Haller P.** A system dynamics approach for assessing the impact of cyber attacks on critical infrastructures // International Journal of Critical Infrastructure Protection. 2015. Vol. 10. P. 3–17.
46. **Yampolskiy M., Horvath P., Koutsoukos X. D., Xue Y., Sztipanovits J.** A language for describing attacks on cyber-physical systems // International Journal of Critical Infrastructure Protection. 2014. Vol. 8. P. 40–52.

47. Liu N., Zhang J., Zhang H., Liu W. Security assessment for communication networks of power control systems using attack graph and MCDM // IEEE Transactions on Power Delivery. 2010. Vol. 25. P. 1492–1500.
48. Wei D., Lu Y., Jafari M., Skare P. M., Rohde K. Protecting smart grid automation systems against cyber attacks // IEEE Transactions on Smart Grid. 2011. Vol. 2. P. 782–795.
49. Ashok A., Hahn A., Govindarasu M. Cyber-physical security of wide-area monitoring, protection and control in a smart grid environment // Journal of Advanced Research. 2014. Vol. 5. P. 481–489.
50. Varshney S., Srivastava L., Pandit M. A parallel computing approach for integrated security assessment of power system // International Journal of Electrical Power & Energy Systems. 2016. Vol. 78. P. 591–599.
51. Mo Y., Chabukswar R., Sinopoli B. Detecting integrity attacks on SCADA systems // IEEE Transactions on Control Systems Technology. 2013. Vol. 22, N. 4. P. 1396–1407.
52. Pasqualetti F., Dorfler F., Bullo F. Attack detection and identification in cyber-physical systems // IEEE Transactions on Automatic Control. 2013. Vol. 58, N. 11. P. 2715–2729.
53. Giani A., Bitar E., Garcia M., McQueen M., Khargonekar P., Poolla K. Smart grid data integrity attacks // IEEE Transactions on Smart Grid. 2013. Vol. 4, N. 3. P. 1244–1253.
54. Wasicek A., Derler P., Lee E. Aspect-oriented Modeling of Attacks in Automotive Cyber-Physical Systems // Proceedings of the 51st Annual Design Automation Conference. 2014. P. 1–6.
55. Cardenas A. A., Amin S., Lin Z. S., Huang Y. L., Huang C. Y., Sastry S. Attacks against process control systems: risk assessment, detection, and response // Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security. 2011. P. 355–366.
56. Mitchell R., Chen I. R. Modeling and analysis of attacks and counter defense mechanisms for cyber physical systems // IEEE Transactions on Reliability. 2015. Vol. 65. P. 350–358.
57. Srivastava A., Morris T. H., Ernster T., Vellaithurai C., Pan S., Adhikari U. Modeling cyber-physical vulnerability of the smart grid with incomplete information // IEEE Transactions on Smart Grid. 2013. Vol. 4. P. 235–245.
58. Friedberg I., Skopik, F. Settanni G., Fiedler R. Combating advanced persistent threats: From network event correlation to incident detection // Computers & Security. 2015. Vol. 48. P. 35–57.
59. Vincent H., Wells L., Tarazaga P., Camelio J. Trojan Detection and Side-Channel Analyses for Cyber-Security in Cyber-Physical Manufacturing Systems // 43rd SME North American Manufacturing Research Conference. 2015. P. 77–85.
60. Yang Q., Chang L., Yu W. On false data injection attacks against kalman filtering in power system dynamic state estimation // International Journal of Security and Communication Networks. 2016. Vol. 9. P. 833–849.
61. Ntalampiras S. Automatic identification of integrity attacks in cyber-physical systems // Expert Systems With Applications. 2016. Vol. 58. P. 164–173.
62. Martins G., Bhatia S., Koutsoukos X., Stouffer K., Tang C., Candell R. Towards a Systematic Threat Modeling Approach for Cyber-physical Systems // Resilience Week (RSW). 2015. P. 114–119.
63. Sanislav T., Miclea L. An agent-oriented approach for cyber-physical system with dependability features // IEEE International Conference on Automation Quality and Testing Robotics. 2012. P. 356–361.
64. Wang P., Xiang Y., Dai B. R. A reliable workflow for cyber-physical system components substitution // International Conference on Computer and Information Technology. 2012. P. 885–891.
65. Hu F., Lu Y., Vasilakos A. V., Hao Q., Ma R., Patil Y., Zhang T., Lu J., Li X., Xiong N. N. Robust cyber-physical systems: Concept, models, and implementation // Future Generation Computer Systems. 2016. Vol. 56. P. 449–475.
66. Yoo H., Shon T. Challenges and research directions for heterogeneous cyber-physical system based on IEC 61850: Vulnerabilities, security requirements, and security architecture // Future Generation Computer Systems. 2016. Vol. 61. P. 128–136.
67. Barnum S., Sastry S., Stankovic J. A. Roundtable: Reliability of embedded and cyberphysical systems // IEEE Security & Privacy. 2010. Vol. 8, N. 5. P. 27–32.
68. Kim K. D., Kumar P. R. Cyber-Physical Systems: A perspective at the centennial // Proceedings of the IEEE. 2012. Vol. 100. P. 1287–1308.
69. Derler P., Lee E. A., Sangiovanni-Vincentelli A. Modeling of Cyber-physical systems // Proceedings of IEEE. 2012. Vol. 100. P. 1–28.
70. Baheti R., Gill H. Cyber-physical systems // The impact of control technology. 2011. Vol. 12. P. 161–166.

R. M. Alguliyev, Full member of Azerbaijan

National Academy of Sciences, DrSc, professor, e-mail: rasim@science.az,

Ya. N. Imamverdiyev, PhD on technical sciences, Assistant Professor, e-mail: yadigar@lan.ab.az,

L. V. Sukhostat, PhD on technical sciences, e-mail: lsuhostat@hotmail.com,

Institute for Information Technologies, Azerbaijan National Academy of Sciences, Baku, Azerbaijan

Cyber-Physical Systems: Basic Concepts and Security Issues

Cyber-physical system is one of the most important and promising technologies. The aim of this study is to select, classify, and analyse the existing researches on the security issues of cyber-physical systems in order to better understand how the security is actually provided to these systems. This paper deals with the state-of-the-art researches in the field of cyber-physical systems security due to the following issues: 1) assessment of the cyberattacks' impact; 2) attacks modeling; 3) intrusion detection; and 4) the development of secure architecture. The basic types of attack and threats classification on cyber-physical systems are considered. The future research directions in the field of cyber-physical systems security are discussed.

Keywords: cyber-physical system, cyber-physical system security, cyber-physical system attacks, cyber-physical system threats, intrusion detection

References

1. Wu G., Sun J., Chen J. A survey on the security of cyber-physical systems, *Control Theory and Technology*, 2016, vol. 14, no. 1, pp. 2–10.
2. NIST, Cyber-Physical Systems. URL: <https://www.nist.gov/el/cyber-physical-systems> (дата обращения: 27.01.17)
3. Rawat D. B., Rodrigues J., Stojmenovic I. *Cyber Physical Systems: From Theory to Practice*. USA: CRC Press, 2015. 570 p.

4. **Jazdi N.** Cyber physical systems in the context of industry 4.0, *Proceedings of the IEEE International Conference on Automation, Quality and Testing, Robotics*, 2014, pp. 1–4.
5. **Lee E. A.** Cyber physical systems: Design challenges, *Proceedings of the 11th International Symposium on Object/Component/Service-Oriented Real-Time Distributed Computing*, 2008, pp. 363–369.
6. **Johansson K. H.** Control of Cyber-physical Systems: Fundamental Challenges and Applications to Transportation Networks, *Proceedings of the 27th International Conference on Architecture of Computing Systems*, 2014, XI p.
7. **Stankovic J. A.** Research directions for the Internet of Things, *IEEE Internet of Things Journal*, 2014, vol. 1, no. 1, pp. 3–9.
8. **Krotofil M., Cardenas A.** Resilience of process control systems to cyberphysical attacks, *Proceedings of the 18th Nordic Conference on Secure IT Systems*, 2013, pp. 166–182.
9. **Kopetz H.** *Real-Time Systems: Design Principles for Distributed Embedded Applications*. USA: Springer, 2011. 378 p.
10. **Hahn A., Thomas R. K., Lozano I., Cardenas A.** A multi-layered and kill-chain based security analysis framework for cyber-physical systems, *International Journal of Critical Infrastructure Protection*, 2015, vol. 11, pp. 39–50.
11. **Krotofil M., Larsen J.** Are you threatening my hazards? *Proceeding of the 9th International Workshop on Security*, 2014, pp. 17–32.
12. **Hills R.** *Common VPN security flaws, White Paper, NTA Monitor*, 2005. URL: <http://www.nta-monitor.com/> (дата обращения: 25.01.17)
13. **Stouffer K., Falco J., Scarfone K.** Guide to Industrial Control Systems (ICS) Security, *NIST Special Publication*, 2011, pp. 800–882.
14. **Majdalawieh M.** *Security Framework for DNP3 and SCADA*, Saarbruken, Germany, VDM Verlag, 2008. 160 p.
15. **Ozansoy C. R., Zayegh A., Kalam A.** Time synchronisation in a IEC 61850 based substation automation system, *Proceedings of the 2008 Australasian Universities Power Engineering Conference*, 2008, pp. 1–7.
16. **Modbus-IDA**, *Modbus Application Protocol Specification V.1.1b*. URL: www.modbus.org/docs/Modbus_Application_Protocol_V1_1b.pdf. (дата обращения: 19.01.17).
17. **Sheth A., Anantharam P., Henson C.** Physical-cyber-social computing: an early 21st century approach, *IEEE Intelligent Systems*, 2013, vol. 28, no. 1, pp. 78–82.
18. **Wang F.-Y.** The Emergence of Intelligent Enterprises: from CPS to CPSS, *IEEE Intelligent Systems*, 2010, vol. 25, no. 4, pp. 85–88.
19. **Zeng J., Yang L. T., Lin M., Ning H., Ma J.** A survey: Cyber-physical-social systems and their system-level design methodology, *Future Generation Computer Systems*, 2016.
20. **Li H., Lai L., Poor H. V.** Multicast routing for decentralized control of cyber physical systems with an application in smart grid, *IEEE Journal on Selected Areas in Communications*, 2012, vol. 30, pp. 1097–1107.
21. **Koubaa A., Andersson B.** A Vision of Cyber-Physical Internet, *Proceedings of the 8th International Workshop on Real-Time Networks*, 2009, pp. 1–6.
22. **Cardenas A. A., Amin S., Sinopoli B., Giani A., Perrig A., Sastry S.** Challenges for securing cyber physical systems, *Workshop on Future Directions in Cyber-physical Systems Security*, 2009, pp. 1–4.
23. **Saltzer J. H., Schroeder M. D.** The protection of information in computer systems, *Proceedings of the IEEE*, 1975, vol. 63, no. 9, pp. 1278–1308.
24. **Avizienis A., Laprie J.-C., Randell B., Landwehr C.** Basic concepts and taxonomy of dependable and secure computing, *IEEE Transactions on Dependable and Secure Computing*, 2004, vol. 1, no. 1, pp. 11–32.
25. **Wang E. K., Ye Y., Xu X., Yiu S. M., Hui L. C. K., Chow K. P.** Security Issues and Challenges for Cyber Physical System, *IEEE/ACM International Conference on Cyber, Physical and Social Computing*, 2010, pp. 733–738.
26. **Sicari S., Rizzardi A., Grieco L. A., Coen-Porisini A.** Security, privacy and trust in Internet of Things: The road ahead, *Computer Networks*, 2015, vol. 76, pp. 146–164.
27. **Xu D., Tu M., Sanford M., Thomas L., Woodraska D., Xu W.** Automated security test generation with formal threat models, *IEEE Transactions on Dependable and Secure Computing*, 2012, vol. 9, no. 4, pp. 525–539.
28. **Xinlan Z., Zhifang H., Guangfu W., Xin Z.** Information Security Risk Assessment Methodology Research: Group Decision Making and Analytic Hierararchy Process, *Second WRI World Congress on Software Engineering*, 2010, pp. 157–160.
29. **Neuman C., Tan K.** Mediating Cyber and Physical Threat Propagation in Secure Smart Grid Architectures, *Second International Conference on Smart Grid Communications*, 2011, pp. 238–243.
30. **Brooks Z.** Hacking Driverless Vehicles, *DEFCON*. URL: <https://www.defcon.org/images/defcon-21/dc-21-presentations/Zoz/DEFCON-21-Zoz-Hacking-Driverless-Vehicles.pdf> (дата обращения: 24.12.16).
31. **Krotofil M., Larsen J., Gollmann D.** The process matters: Ensuring data veracity in cyber-physical systems, *Proceedings of the 10th ACM Symposium on Information, Computer and Communications Security*, 2015, pp. 133–144.
32. **Djouadi S. M., Melin A. M., Ferragut E. M., Laska J. A., Dong J.** Finite Energy and Bounded Actuator Attacks on Cyber-Physical Systems, *Proceedings of the IEEE European Control Conference*, 2015, pp. 3659–3664.
33. **Singhal A.** *Data Warehousing and Data Mining Techniques for Cyber Security*, USA: Springer Science + Business Media. 2007. 159 p.
34. **Mitchell R., Chen I. R.** Effect of intrusion detection and response on reliability of cyber physical systems, *IEEE Transactions on Reliability*, 2013, vol. 62, no. 1, pp. 199–210.
35. **Cardenas A. A., Amin S., Sastry S.** Research challenges for the security of control systems, *Proceedings of the 3rd conference on Hot Topics in Security*, 2008, pp. 1–6.
36. **K. Wan, Man K. L., Hughes D.** Specification, Analyzing Challenges and Approaches for Cyber-Physical Systems (CPS), *Engineering Letters*, 2010, vol. 18, no. 3, pp. 308–315.
37. **Sha L., Meseguer J.** Design of complex cyber physical systems with formalized architectural patterns, *Software-Intensive Systems and New Computing Paradigms*. Berlin: Springer-Verlag, 2008, pp. 92–100.
38. **Karnouskos S.** Stuxnet worm impact on industrial cyber-physical system security, *Proceedings of the 37th Annual Conference of the IEEE Industrial Electronics Society*, 2011, pp. 4490–4494.
39. **Collins S., McCombie S.** Stuxnet: the emergence of a new cyber weapon and its implications, *Journal of Policing, Intelligence and Counter Terrorism*, 2012, vol. 7, no. 1, pp. 80–91.
40. **Wu W., Kang R., Li Z.** Risk Assessment Method for Cyber Security of Cyber Physical Systems, *Proceedings of 2015 the 1st International Conference On Reliability Systems Engineering*, 2015, pp. 523–527.
41. **Orojloo H., Abdollahi Azgomi M.** A method for evaluating the consequence propagation of security attacks in cyber-physical systems, *Future Generation Computer Systems*, 2017, p. 57–71.
42. **Krotofil M., Cardenas A. A., Larsen J., Gollmann D.** Vulnerabilities of cyber-physical systems to stale data-determining the optimal time to launch attacks, *International Journal of Critical Infrastructure Protection*, 2014, vol. 7, pp. 213–232.
43. **Huang Y. L., Cardenas A. A., Amin S., Lin Z. S., Tsai H. Y., Sastry S.** Understanding the physical and economic consequences of attacks on control systems, *International Journal of Critical Infrastructure Protection*, 2009, vol. 2, pp. 73–83.
44. **Genge B., Siaterlis C., Hohenadel M.** Impact of network infrastructure parameters to the effectiveness of cyber attacks against industrial control systems, *International Journal of Computers, Communications & Control*, 2014, vol. 7, pp. 674–687.
45. **Genge B., Kiss I., Haller P.** A system dynamics approach for assessing the impact of cyber attacks on critical infrastructures, *International Journal of Critical Infrastructure Protection*, 2015, vol. 10, pp. 3–17.
46. **Yampolskiy M., Horvath P., Koutsoukos X. D., Xue Y., Sztiapanovits J.** A language for describing attacks on cyber-physical systems

tems, *International Journal of Critical Infrastructure Protection*, 2014, vol. 8, pp. 40–52.

47. Liu N., Zhang J., Zhang H., Liu W. Security assessment for communication networks of power control systems using attack graph and MCDM, *IEEE Transactions on Power Delivery*, 2010, vol. 25, pp. 1492–1500.

48. Wei D., Lu Y., Jafari M., Skare P. M., Rohde K. Protecting smart grid automation systems against cyber attacks, *IEEE Transactions on Smart Grid*, 2011, vol. 2, pp. 782–795.

49. Ashok A., Hahn A., Govindarasu M. Cyber-physical security of wide-area monitoring, protection and control in a smart grid environment, *Journal of Advanced Research*, 2014, vol. 5, pp. 481–489.

50. Varshney S., Srivastava L., Pandit M. A parallel computing approach for integrated security assessment of power system, *International Journal of Electrical Power & Energy Systems*, 2016, vol. 78, pp. 591–599.

51. Mo Y., Chabukswar R., Sinopoli B. Detecting integrity attacks on SCADA systems, *IEEE Transactions on Control Systems Technology*, 2013, vol. 22, no. 4, pp. 1396–1407.

52. Pasqualetti F., Dorfler F., Bullo F. Attack detection and identification in cyber-physical systems, *IEEE Transactions on Automatic Control*, 2013, vol. 58, no. 11, pp. 2715–2729.

53. Giani A., Bitar E., Garcia M., McQueen M., Khargonekar P., Poolla K. Smart grid data integrity attacks, *IEEE Transactions on Smart Grid*, 2013, vol. 4, no. 3, pp. 1244–1253.

54. Wasicek A., Derler P., Lee E. Aspect-oriented Modeling of Attacks in Automotive Cyber-Physical Systems, *Proceedings of the 51st Annual Design Automation Conference*, 2014, pp. 1–6.

55. Cardenas A. A., Amin S., Lin Z. S., Huang Y. L., Huang C. Y., Sastry S. Attacks against process control systems: risk assessment, detection, and response, *Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security*, 2011, pp. 355–366.

56. Mitchell R., Chen I. R. Modeling and analysis of attacks and counter defense mechanisms for cyber physical systems, *IEEE Transactions on Reliability*, 2015, vol. 65, pp. 350–358.

57. Srivastava A., Morris T. H., Ernster T., Vellaithurai C., Pan S., Adhikari U. Modeling cyber-physical vulnerability of the smart grid with incomplete information, *IEEE Transactions on Smart Grid*, 2013, vol. 4, pp. 235–245.

58. Friedberg I., Skopik, F. Settanni G., Fiedler R. Combating advanced persistent threats: From network event correlation to incident detection, *Computers & Security*, 2015, vol. 48, pp. 35–57.

59. Vincent H., Wells L., Tarazaga P., Camelio J. Trojan Detection and Side-Channel Analyses for Cyber-Security in Cyber-Physical Manufacturing Systems, *43rd SME North American Manufacturing Research Conference*, 2015, pp. 77–85.

60. Yang Q., Chang L., Yu W. On false data injection attacks against kalman filtering in power system dynamic state estimation, *International Journal of Security and Communication Networks*, 2016, vol. 9, pp. 833–849.

61. Ntalampiras S. Automatic identification of integrity attacks in cyber-physical systems, *Expert Systems With Applications*, 2016, vol. 58, pp. 164–173.

62. Martins G., Bhatia S., Koutsoukos X., Stouffer K., Tang C., Candell R. Towards a Systematic Threat Modeling Approach for Cyber-physical Systems, *Resilience Week (RSW)*, 2015, pp. 114–119.

63. Sanislav T., Miclea L. An agent-oriented approach for cyber-physical system with dependability features, *IEEE International Conference on Automation Quality and Testing Robotics*, 2012, pp. 356–361.

64. Wang P., Xiang Y., Dai B. R. A reliable workflow for cyber-physical system components substitution, *International Conference on Computer and Information Technology*, 2012, pp. 885–891.

65. Hu F., Lu Y., Vasilakos A. V., Hao Q., Ma R., Patil Y., Zhang T., Lu J., Li X., Xiong N. N. Robust cyber-physical systems: Concept, models, and implementation, *Future Generation Computer Systems*, 2016, vol. 56, pp. 449–475.

66. Yoo H., Shon T. Challenges and research directions for heterogeneous cyber-physical system based on IEC 61850: Vulnerabilities, security requirements, and security architecture, *Future Generation Computer Systems*, 2016, vol. 61, pp. 128–136.

67. Barnum S., Sastry S., Stankovic J. A. Roundtable: Reliability of embedded and cyberphysical systems, *IEEE Security & Privacy*, 2010, vol. 8, no. 5, pp. 27–32.

68. Kim K. D., Kumar P. R. Cyber-Physical Systems: A perspective at the centennial, *Proceedings of the IEEE*, 2012, vol. 100, pp. 1287–1308.

69. Derler P., Lee E. A., Sangiovanni-Vincentelli A. Modeling of Cyber-physical systems, *Proceedings of IEEE*, 2012, vol. 100, pp. 1–28.

70. Baheti R., Gill H. Cyber-physical systems, *The impact of control technology*, 2011, vol. 12, pp. 161–166.

ГЕОИНФОРМАЦИОННЫЕ СИСТЕМЫ GEOINFORMATION SYSTEMS

УДК 004.4, 528.8

З. З. Ружичка^{1, 2}, инженер, e-mail: tramtara@seznam.cz,

К. В. Музалевский¹, канд. физ.-мат. наук, и.о. зав. лаб.,

О. Э. Якубайлик³, канд. физ.-мат. наук, ст. науч. сотр., e-mail: oleg@icm.krasn.ru,

Е. Г. Швецов⁴, канд. техн. наук, науч. сотр., eugeneshvetsov@mail.ru

¹ Институт физики им. Л. В. Киренского СО РАН, ФИЦ КНЦ СО РАН, Красноярск

² Сибирский государственный аэрокосмический университет

имени академика М. Ф. Решетнева, Красноярск

³ Институт вычислительного моделирования СО РАН, ФИЦ КНЦ СО РАН, Красноярск

⁴ Институт леса им. В. Н. Сукачева СО РАН, ФИЦ КНЦ СО РАН, Красноярск

Информационно-технологическое обеспечение задач обработки и представления радиометрических спутниковых данных

Рассматриваются проблемы и методы, программные средства для обработки и представления радиометрических спутниковых данных SMOS, GCOM-W1, MODIS, используемых в исследованиях различных характеристик почвы (влажность, температура). Создана база геопространственных данных, программы и прикладные сервисы для различных операций над ними. Разработанный комплекс программ обеспечивает решение всех необходимых задач автоматизации научных исследований.

Ключевые слова: радиометрические данные, SMOS, влажность почвы, обработка спутниковых данных, прикладной сервис

Введение

В условиях малой продолжительности светлого времени суток, высокой вероятности закрытости облачностью и других природных и климатических особенностей значительных по размеру северных территорий России космическая радиолокация становится основным средством, позволяющим обеспечить регулярный и высокоинформативный мониторинг в интересах решения различных научных и производственных задач. В частности, космические радиолокационные методы мониторинга на основе данных микроволнового диапазона более 20 лет успешно применяют для определения температурно-влажностных характеристик атмосферы и подстилающей поверхности, дистанционного зондирования процессов оттаивания/замерзания деятельного слоя почв арктических территорий; с помощью радарных измерений создают пространственно-временные карты талых и мерзлых почв северных территорий планеты. Содержанием проводимых научных исследований является разработка и реализация методов и алгоритмов обработки данных спутникового мониторинга, обеспечивающие получение тематических информационных

продуктов на основе исходных данных дистанционного зондирования (ДЗ).

Исследования, проводимые коллективом авторов настоящей статьи, связаны с разработкой и верификацией алгоритмов восстановления влажности и температуры почвы из радарных и радиометрических измерений яркостной температуры поверхности Земли на основе реалистичной модели комплексной диэлектрической проницаемости тундровой почвы, физико-математического моделирования радиотеплового излучения. При проведении исследований используют спутниковые данные микроволнового диапазона частот, получаемые с современных космических аппаратов GCOM-W1, SMOS, SMAP, и другие данные. Особое внимание уделяется данным космического аппарата SMOS, для использования которых создана одночастотная температурно-зависимая модель комплексной диэлектрической проницаемости органической почвы. Как показала практика, возникающие в процессе обработки этих данных технические сложности и вычислительные затраты могут быть в значительной степени уменьшены за счет промежуточного их преобразования, учитывающего последующие ме-

тоды анализа. Разработанный комплекс специализированных программно-технологических средств для предварительной обработки радиометрических спутниковых данных обеспечивает решение рассматриваемых задач физико-математического моделирования, позволяет значительно повысить эффективность исследований.

Данные SMOS и программное обеспечение для их обработки

Запущенный в 2009 г. спутник Европейского космического агентства SMOS (Soil Moisture and Ocean Salinity) предназначен для получения информации по влажности почв и солености океана. Он оснащен интерференционным радиометром MIRAS (Microwave Imaging Radiometer with Aperture Synthesis), работающем в диапазоне частот 1,400...1,427 ГГц (L-диапазон), который формирует данные в глобальном масштабе с пространственным разрешением 30–50 км. Ширина полосы съемки составляет 600 км, ее повторяемость — 3 дня; контур радиометрического снимка MIRAS имеет характерную гексагональную форму.

Потребителям доступны данные SMOS нескольких типов: семейство данных 1-го уровня (L1C) — исходные данные по яркостной температуре, базовые процедуры обработки, а также данные 2-го уровня (L2), включающие влажность почвы и соленость океана, различные дополнительные геофизические параметры. Основным источником информации в настоящее время является портал Европейского космического агентства (ЕКА) ESA SMOS Online Dissemination Service, с помощью которого можно получить данные через интерактивные веб-сервисы, *http/ftp*-протоколы. В некоторых случаях целесообразно воспользоваться информацией национальных центров обработки данных Франции (Centre Aval de Traitement des Données SMOS — CATDS) и Испании (SMOS Barcelona Expert Centre on Radiometric Calibration and Ocean Salinity — SMOS-BEC), которые формируют и распространяют продукты 3-го (L3) и 4-го (L4) уровня обработки. Оперативные данные по влажности почвы (L2) также доступны через сервисы Европейской ассоциации спутниковой метеорологии EUMETSAT (в течение 4 ч после съемки).

Формат данных SMOS соответствует стандартам Европейского космического агентства — вся информация содержится в двух составных частях — метаданные (*.HDR файл заголовка в виде XML) и собственно данные (бинарный файл *.DBL — data block). Для чтения, интерпретации этих данных необходимо специализированное программное обеспечение — используемые форматы не поддерживаются популярными пакетами программ, распространенными геоинформационными системами.

Основное программное обеспечение для чтения и обработки данных SMOS разрабатывается при поддержке ЕКА:

- BEAM (VISAT) / SMOS ToolBox (разработка приостановлена в середине 2016 г.) — доступный в исходных кодах набор программных инструментов и платформа для разработчиков программного обеспечения, предназначенные для визуализации, анализа и обработки растровых данных дистанционного зондирования [1]. Функционал этого программного обеспечения сейчас перенесен в представленный ниже набор инструментов SNAP (SeNtinel Application Platform).
- SMOS Toolbox / SNAP — последняя версия набора программных инструментов для обработки данных SMOS, основанная на новой архитектуре свободного программного обеспечения ЕКА, представленная в 2016 г. Комплекс программ SNAP включает обновленный набор инструментов для обработки спутниковых данных различных миссий ЕКА (Sentinel 1/2/3, ERS-Envisat и др.), а также переработанные программные модули из ряда ранее разработанных программных библиотек и компонентов — BEAM, NEST, Orfeo Toolbox и проч. [2].
- ESOV (The Earth observation Swath and Orbit Visualization tool) — вспомогательная программа для формирования списка файлов для заданной территории и диапазона дат съемки с данными различных спутниковых миссий ЕКА (Envisat, ERS-1, ERS-2, Cryosat, SMOS, Metop, Sentinel-1/2/3 и др.) в целях их последующей загрузки в пакетном режиме.
- SMOS NetCDF Conversion Tool — утилита (консольное приложение) для преобразования данных SMOS в формат NetCDF.
- MATLAB READ API — интерфейс для прямого чтения данных SMOS из прикладной программы MATLAB.
- SMOS Global Mapping Tool (GMT) — программное обеспечение для визуализации данных SMOS на единой глобальной географической карте.
- HDFViewer — программа предназначена для визуализации любых данных в HDF-формате, в том числе данных SMOS.
- SMOSView — программа предоставляет средства декодирования и визуализации данных SMOS, экспорта данных в различные форматы.

Перечисленное программное обеспечение предназначено для различных аспектов работы с данными SMOS. Несмотря на многочисленные достоинства представленных программных средств, следует обратить внимание на то, что в настоящее время в перечне рекомендуемых ЕКА к использованию приложений нет инструментов для обработки больших объемов данных и формирования собственной базы геопространственных данных. Для иллюстрации

необходимости такого рода инструментов рассмотрим в качестве примера нижеследующую ситуацию. Пусть нам необходимо проанализировать временной ряд данных зависимости яркостной температуры от угла падения для относительно небольшого участка размером 100 × 100 км. Данные доступны в виде набора бинарных файлов (.DBL), каждый из которых занимает около 200 Мбайт места на диске, при этом он содержит информацию на всю территорию съемки, с шириной охвата около 1000 км. Нам необходимо из каждого такого исходного файла извлечь относительно небольшую порцию данных, размером на два порядка меньше суммарного объема файла. Таким образом, для извлечения и последующего анализа нескольких мегабайт данных необходима обработка нескольких гигабайт входной информации. В этом случае целесообразность создания собственного производного набора данных, в котором содержится только необходимая информация, становится очевидной. Уменьшение размера хранимых радиометрических спутниковых данных на территории полуострова Ямал за период с января 2011 по июль 2015 г., достигнутое за счет применения разработанного программно-технологического обеспечения, проиллюстрировано в таблице. Здесь стоит также отметить, что приведенное здесь уменьшение размера хранимых данных — важный, но не единственный результат работы разработанного программного обеспечения. Предложенная новая структура хранения радиолокационных данных и сервисы для доступа к ним учитывают потребности решаемых задач физико-математического моделирования, упрощают и ускоряют их реализацию. Актуальности разработки специализированных программных инструментов обработки радиометрических данных, оптимизации хранения данных, формированию оптимальных структур данных SMOS посвящен ряд работ [3—5].

При формировании базы геопространственных данных SMOS одним из основных вопросов является способ хранения радиометрических данных SMOS 1-го уровня обработки (L1C). Для каждой точки с заданными географическими координатами необходимо хранить информацию о яркостной температуре для четырех поляризации (горизонтальной, вертикальной и двух кросс-поляризации) и углах падения от 0 по 65° для каждой из них. Спутниковые данные могут содержать эту информацию с разной степенью детализации. Например, формируемый массив данных для заданной точки

Изменение размера исходных данных SMOS в результате их преобразования в специализированную базу данных

Тип данных	Число файлов	Размер исходных данных SMOS, Гбайт	Размер сформированной базы данных, Гбайт
SMOS L2 SM	4697	117	3,3
SMOS L1 BT	4697	2583	284

на местности (участка гексагональной формы) и поляризации может содержать не более 10 измерений яркостной температуры для разных значений угла падения, а может быть и более 60, как на рис. 1 (см. третью сторону обложки).

В некоторых случаях целесообразно использовать данные SMOS 3-го уровня обработки (L3), которые предоставляет французский центр обработки данных CATDS в формате NetCDF. Это усредненные за сутки значения яркостной температуры для участков местности фиксированного размера по широте и долготе; по углу падения выполнена агрегация и интерполяция с шагом в 5°. Сформированный набор данных имеет фиксированное число слоев и является универсальным. Преимущество его применения также обусловлено удобным сервисом получения данных, обеспечивающим извлечение данных на заданную территорию. Благодаря широкой поддержке формата NetCDF в различных современных ГИС и пакетах программ для математического моделирования использование таких данных, как правило, не вызывает затруднений.

Неизбежным недостатком стандартного формата данных с ячейками фиксированного размера являются большие погрешности в данных для высокоширотных территорий, где имеет место значительная разница в метрических размерах ячеек сетки по широте/долготе.

Актуальная версия программных инструментов SMOS Toolbox / SNAP также предоставляет средства формирования наборов данных с заданными характеристиками (для определяемой пользователем территории, типов поляризации и углов падения), их последующей конвертации в формат NetCDF, обеспечивает решение задач предварительной подготовки данных в полном объеме. Следует рассматривать данные технологии в качестве основных для обработки данных SMOS, несмотря на то, что их использование, как уже отмечалось выше, требует значительных ресурсов и вычислений.

Рассматривая различные аспекты представления и анализа данных о яркостной температуре и влажности почвы, следует также обратить внимание на то, что основанные на них исследования не ограничиваются спутником SMOS. В настоящее время есть еще несколько спутников, которые можно использовать: японский GCOM-W1; американский SMAP и др. Очевидная идея формирования сводного массива информации на основе данных, поступающих с различных спутников, приводит к необходимости разработки базы геопространственных данных на основе технологий СУБД. Подобная база данных должна выступать в качестве хранилища геопозиционированной информации, обеспечивать выполнение обычных и пространственных запросов к данным, быть по возможности совместимой с существующим программным обеспечением ГИС. Наряду с самой базой геопростран-

ственных данных должны быть разработаны программные инструменты для работы с ней — загрузки различных спутниковых данных, их обработки, выгрузки сводной информации в популярные форматы ГИС и др. — без этих средств рассматриваемая геопространственная база данных не будет иметь смысла. Также должна быть обеспечена совместимость с популярными системами и протоколами информационного взаимодействия.

Структура программно-технологического обеспечения

Для информационно-аналитического обеспечения исследований влажности почвы и других характеристик поверхности Земли по радиометрическим спутниковым данным создан комплекс программных средств, в состав которого входят как сторонние, так и собственные оригинальные программные компоненты. Реализованы модули загрузки, чтения/записи спутниковых данных, алгоритмы их

обработки и создания тематических продуктов, экспорта данных в форматы популярных ГИС и обеспечения прямого доступа к данным на основе картографических веб-сервисов (рис. 2). Разработка была выполнена на основе свободного и открытого программного обеспечения ГИС и баз данных [6–9].

Ядро системы реализовано на языке программирования MATLAB. На нем разработана библиотека функций и методов, обеспечивающая запуск сторонних программ, подключение внешних библиотек. В частности, библиотека PyMODIS используется для загрузки данных MODIS; программа WinSCP предоставляет сервисы загрузки данных SMOS 1-го и 2-го уровня (L1/L2) и GCOM-W1; приложение SmosDataExtractor решает задачи первичной обработки исходных данных SMOS L1/L2; программные модули GDAL/OGR выполняют преобразования растровых и векторных данных; СУБД PostgreSQL/PostGIS обеспечивает поддержку базы геопространственных данных.

Разработанная система включает средства получения исходных данных из различных источников, сервисы первичной обработки данных, создания и наполнения базы геоданных, их визуализации, экспорта в популярные форматы геоинформационных систем.

Открытая архитектура разработки предполагает возможность расширения ее программных элементов для различных задач. Например, можно воспользоваться картографическими веб-сервисами для визуализации информации из базы геоданных в какой-либо современной ГИС, или установить и настроить картографический веб-сервер со службами каталога геопространственных данных, сформировать тематический геопортал по рассматриваемой проблеме на основе имеющегося набора данных [10].

Информационные потоки данных разработанной системой представлены на рис. 3. "Пользователь" формирует запрос к данным — выбирает с помощью набора координат интересующий его участок местности (область интереса), указывает другие параметры. Выполняются первоначальная обработка, получение данных из внешних источников, формирование локальной базы геопространственных данных. Отличительной особенностью разработки по сравнению с аналогами является ее универсальность — возможность получения данных различных спутниковых систем, унифицированные сервисы первичной обработки, представления и экспорта данных, в том числе на основе международных геопространственных стандартов Консорциума OGC. Стоит отметить также, что в системе предусмотрена возможность интеграции, совместной обработки

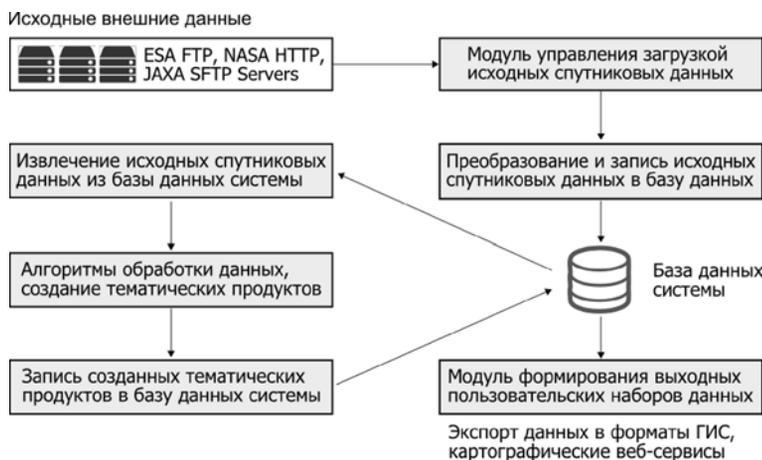


Рис. 2. Структура разработанного программно-технологического обеспечения системы информационной поддержки исследований влажности почвы по спутниковым данным

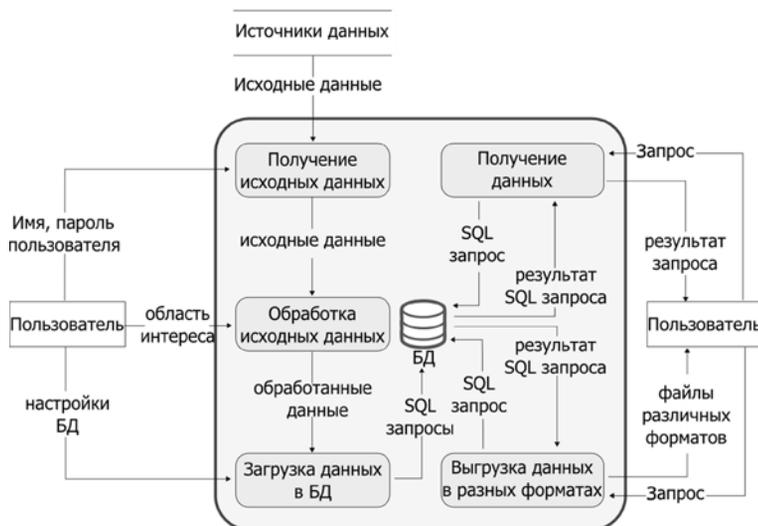


Рис. 3. Информационные потоки данных разработанной системы

и анализа данных, поступающих с разных спутниковых систем.

После загрузки данных можно приступить к их анализу, визуализации и обработке, используя разработанные программные компоненты, сервисы и пользовательские интерфейсы. С помощью языка структурированных запросов можно формировать подмножества данных, необходимые для исследовательских задач.

В системе предусмотрена настройка различных параметров с помощью конфигурационных файлов, которые упрощают решение типовых задач пользователей, таких как разграничение прав доступа и подключение к базе данных, выбор и визуализация информации из базы геопространственных данных, их преобразование и экспорт в сторонние форматы.

Система предполагает работу с радиометрическими данными, получаемыми с разных спутников, каждый из которых имеет собственную структуру хранения информации. Для объединения поступающей с различных спутников исходной информации в единой базе данных был разработан "универсальный" формат хранения радиометрических данных, в который можно конвертировать все актуальные на данный момент спутниковые данные — в нем учтены все возможные вариации различных параметров. Например, радиометрические данные SMOS определены для одной фиксированной частоты и различных углов падения. В отличие от них спутниковые данные GCOM-W1 регистрируются для нескольких различных частот и одного фиксированного угла падения. В разработанной системе для каждого источника исходных данных сформирован отдельный сервис со своим набором характеристик, с помощью которого можно загружать данные в систему. Характеристики сервисов определяют подмножество извлекаемой из исходных данных информации, необходимой для хранения в базе данных, использования для физико-математического моделирования в дальнейшем.

При этом различия в исходных данных ограничивают возможности аналитической обработки. Например, для данных SMOS можно построить график угловой зависимости яркостной температуры для выбранного участка местности на заданную дату, а для данных GCOM-W1 такой график построить нельзя.

Приведенные ниже основные модули разработанного программно-технологического обеспечения реализуют различные функции, связанные с загрузкой, хранением и обработкой геопространственных радиометрических данных.

- Интегрированные сервисы исходных спутниковых данных — программные компоненты, предназначенные для загрузки из внешних

источников и различных операций над используемыми спутниковыми данными — SMOS, GCOM-W1, MODIS и т. д. Для каждого типа (источника) данных создан отдельный сервис, имеющий собственный набор свойств и методов, различных процедур предварительной обработки информации.

- Единая подсистема настройки параметров управления данными реализует унифицированный подход к настройке и конфигурации программного комплекса, декларации различных параметров (констант, режимов вычислений, авторизации при подключении к базе данных, значений по умолчанию), необходимых для обработки геопространственных данных. Реализована возможность работы с динамически настраиваемым классом текущих параметров программного окружения и конфигурационными файлами.
- Сервисы операций с базой геопространственных данных — построение таблиц, триггеров, индексов и проч.; шаблоны SQL-запросов и специализированные методы для извлечения тематических данных разных спутников. Также предусмотрен набор операций по обслуживанию базы данных.

Основное внимание на данном этапе работ было сосредоточено на разработке трех интегрированных сервисов для различных спутниковых данных: SMOSService, GCOMService и MODISService. Они самостоятельно инициируют выполнение модулей, необходимых для различных задач: получения исходных данных; первичной обработки и загрузки в базу геопространственных данных, извлечение данных из базы данных. Пример обращения к сервису и последовательность выполнения этапов работы — от получения исходных данных по влажности почвы до формирования результирующих SHP-файлов геоинформационной системы ArcGIS — показаны на диаграмме (рис. 4).

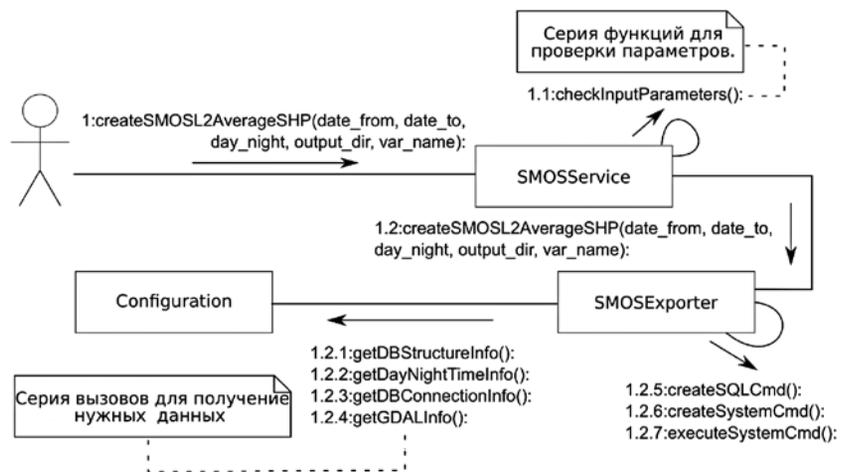


Рис. 4. Этапы выполнения интегрированного сервиса обработки спутниковых данных SMOS в разработанной системе

Заключение

Разработанный комплекс программных средств для информационного обеспечения исследований различных характеристик природной среды по спутниковым данным SMOS, GCOM-W1 и MODIS MOD11A1/MYD11A1 зарекомендовал себя как эффективный инструмент обработки и визуализации данных на региональном уровне. Созданные геопространственная база данных и программное обеспечение позволяют проводить одновременную обработку перечисленных выше данных различных спутниковых систем, использовать полученную информацию в исследованиях протекающих физических процессов, математическом моделировании. Применение в представленной разработке открытых международных стандартов Открытого геопространственного консорциума (OGC) на картографические геопространственные данные значительно упрощает доступ к информации по рассматриваемой теме, расширяет аудиторию ее потенциальных пользователей.

Следует также особо отметить возможности разработанной системы в обработке спутниковых данных SMOS 1-го и 2-го уровней (L1C и L2 SM) — созданное программное обеспечение позволяет значительно сократить вычислительные затраты за счет предварительной обработки исходных данных, а также структуры хранения формируемых массивов информации, учитывающей потребности решаемых исследовательских задач. Разработаны средства анализа временных рядов радиометрических спутниковых данных, инструменты исследования зависимостей различных величин.

Результатом внедрения системы стало значительное повышение эффективности исследований различных характеристик природной среды на основе спутниковых радиометрических данных, таких как определение влажности или температуры почвы. Она проявляется прежде всего в сокращении времени обработки и анализа данных за счет уменьшения их объема, снижения вычислительной нагрузки, возможности работы с информацией в режиме прямого доступа, т. е. без необходимости ее промежуточной конвертации. По экспертным оценкам совокупность перечисленных факторов привела к уменьшению времени обработки данных на типовых рассматриваемых задачах в 3—5 раз. Данная разработка широко применяется в повседневных исследованиях в лаборатории радиофизики дистанционного зондирования Института физики им. Л. В. Киренского СО РАН [11—15].

Некоторые результаты обработки данных, подготовленные с помощью рассматриваемого программного обеспечения, представлены на геопортале Института вычислительного моделирования СО РАН [16]. Сервисы геопортала предоставляют возможности интерактивной веб-картографии, пря-

мого доступа к данным на основе открытых международных стандартов.

Исследование выполнено при финансовой поддержке РФФИ и Правительства Красноярского края в рамках научного проекта № 16-45-242162.

Список литературы

1. **Quast R., Fomferra N., Block T.** SMOS User Support Tools: SMOS-Box and Cal/Val Data Distribution // Proc. of ESA Living Planet Symposium. ESA, 2010. С. 1—3.
2. **ESA Scientific Toolbox Exploitation Platform / The European Space Agency.** URL: <http://step.esa.int/> (дата обращения: 04.12.2016).
3. **Corbella I., Torres F., Duffo N., Gonzalez V., Camps A., Vall-Llosera M.** Fast processing tool for SMOS data // IEEE Geoscience and Remote Sensing Symposium, IGARSS 2008, Boston, MA, USA. 2008. Vol. 2. P. 11521155.
4. **Blankenship C. B., Case J. L., Zavodsky B. T., & Crosson W. L.** Assimilation of SMOS retrievals in the land information system // IEEE Transactions on Geoscience and Remote Sensing. 2016. Vol. 54, N. 11. P. 6320—6332.
5. **Talone M., Portabella M., Martínez J., & González-Gambau V.** About the optimal grid for SMOS level 1C and level 2 products // IEEE Geoscience and Remote Sensing Letters. 2015. Vol. 12, N. 8. P. 1630—1634.
6. **Steiniger S., Hunter A. J. S.** Free and open source GIS software for building a spatial data infrastructure // Geospatial Free and Open Source Software in the 21st Century: Proceedings of the first Open Source Geospatial Research Symposium. Berlin Heidelberg: Springer, 2012. P. 247—261.
7. **Corti P., Mather S. V., Kraft T. J.** Working with Raster Data. In PostGIS Cookbook: Packt Publishing, 2014. С. 173—213.
8. **Miaomiao S., Wenwen L., Bin Z.** Spatiotemporal data representation and its effect on the performance of spatial analysis in a cyberinfrastructure environment — A case study with raster zonal analysis // Computers & Geosciences. 2016. Vol. 87. P. 11—21.
9. **Agosto E.** Vector—raster server-side analysis: a PostGIS benchmark // Applied Geomatics. 2013. Vol. 5, N. 2. P. 177—184.
10. **Ruzicka Z., Muzalevskiy K. V.** Web-GIS application for temperature and moisture monitoring of the arctic tundra soils based on SMOS data // Региональные проблемы дистанционного зондирования Земли: материалы международной науч. конф. / науч. ред. Е. А. Ваганов. Красноярск: Сиб. федер. ун-т, 2014. С. 287—290.
11. **Ruzicka Z., Muzalevskiy K. V.** Database approach to develop and validate the Soil Moisture and temperature retrieval algorithm using SMOS radiometric data // 2015 International Siberian Conference on Control and Communications (SIBCON). Proceedings. Omsk: Omsk State Technical University, 2015. P. 1—5.
12. **Mironov V. L., Muzalevskiy K. V., Ruzicka Z.** Retrieving Profile Temperatures in a Frozen Topsoil Near the TFS, Alaska, Based on SMOS Brightness Temperatures at the 1.4 GHz Frequency // IEEE Transactions on Geoscience and Remote Sensing. 2016. Vol. PP, Is. 99, no. 7557037.
13. **Muzalevskiy K. V., Ruzicka Z.** Retrieving Soil Temperature at a Test Site on the Yamal Peninsula Based on the SMOS Brightness Temperature Observations // IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing. 2016. Vol. 9, Is. 6. P. 2468—2477.
14. **Muzalevskiy K. V., Mikhailov M. I., Mironov V. L., Ruzicka Z.** Retrieving Soil Moisture and temperature using SMOS observations at a test site in the Yamal Peninsular // 2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS). IEEE. 2016. P. 4932—4935.
15. **Muzalevskiy K. V., Ruzicka Z., Kosolapova L. G., Mironov V. L.** Temperature dependence of SMOS/MIRAS, GCOM-W1/AMSR2 brightness temperature and ALOS/PALSAR radar backscattering at arctic test sites // Progress in Electromagnetic Research Symposium (PIERS). IEEE. 2016. PP. 3578—3582.
16. **Геопортал** Института вычислительного моделирования СО РАН. URL: <http://gis.krasn.ru/> (дата обращения: 04.12.2016).

Z. Z. Ruzicka^{1, 2}, Engineer, **K. V. Muzalevskiy**¹, Acting Head of the Laboratory,
O. E. Yakubailik³, Senior Researcher, e-mail: oleg@icm.krasn.ru, **E. G. Shvetsov**⁴, Researcher
¹Kirensky Institute of Physics, Siberian Branch of Russian Academy of Sciences, Krasnoyarsk
²Siberian State Aerospace University, Krasnoyarsk

³Institute of Computational Modelling, Siberian Branch of Russian Academy of Sciences, Krasnoyarsk

⁴V. N. Sukachev Institute of Forest, Siberian Branch of Russian Academy of Sciences, Krasnoyarsk

Information Technological Support of Tasks of Processing and Submission of Radiometric Satellite Data

Problems, methods and software tools for processing and presentation of SMOS, GCOM-W1 and MODIS radiometric satellite data are discussed. These data are used in studies of different soil characteristics, such as soil moisture and temperature. The main attention is paid to SMOS satellite data and the discussion different characteristics of the software to handle them. Article presents a list of existing software toolboxes for various operations on these data.

A database of geospatial data along with software and applied services for a variety of operations on them have been created. Efficient data storage structure that provides a significant increase in computational performance for a number of tasks has been proposed. A number of software functions have been implemented for downloading, storing and processing of geospatial radiometric data. The developed system provides setting of various parameters through configuration files, which simplify common user tasks, such as granting access rights for database connection, selection and visualization of information from geospatial database, conversion and export data to third party formats.

Using of open international mapping standards of Open Geospatial Consortium (OGC) for geospatial data greatly simplifies access to information on the topic, expanding the audience of potential users. The software tools developed can be used for solving different problems arising in scientific investigations. It is widely used in everyday research in the Laboratory of Radiophysics of the Remote Sensing, Kirensky Institute of Physics, Siberian Branch of Russian Academy of Sciences.

Keywords: radiometric data, SMOS, soil moisture, brightness temperature, satellite data processing, software service

References

1. **Quast R., Fomferra N., Block T.** SMOS User Support Tools: SMOS-Box and Cal/Val Data Distribution, *Proceedings of ESA Living Planet Symposium. ESA*, 2010, pp. 1–3.
2. **ESA** Scientific Toolbox Exploitation Platform, The European Space Agency. URL: <http://step.esa.int/> (accessed 04.12.2016).
3. **Corbella I., Torres F., Duffo N., Gonzalez V., Camps A., Vall-Llosera M.** Fast processing tool for SMOS data, *IEEE Geoscience and Remote Sensing Symposium, IGARSS 2008*, Boston, MA, USA, 2008, vol. 2, p. 11521155.
4. **Blankenship C. B., Case J. L., Zavodsky B. T., Crosson W. L.** Assimilation of SMOS retrievals in the land information system, *IEEE Transactions on Geoscience and Remote Sensing*, 2016, vol. 54, no. 11, pp. 6320–6332.
5. **Talone M., Portabella M., Martínez J., González-Gambau V.** About the optimal grid for SMOS level 1C and level 2 products, *IEEE Geoscience and Remote Sensing Letters*, 2015, vol. 12, no. 8, pp. 1630–1634.
6. **Steiniger S., Hunter A. J. S.** Free and open source GIS software for building a spatial data infrastructure: Geospatial Free and Open Source Software in the 21st Century, *Proceedings of the first Open Source Geospatial Research Symposium: Springer Berlin Heidelberg*, 2012, pp. 247–261.
7. **Corti P., Mather S. V., Kraft T. J.** Working with Raster Data. In *PostGIS Cookbook: Packt Publishing*, 2014, pp. 173–213.
8. **Miaomiao S., Wenwen L., Bin Z.** Spatiotemporal data representation and its effect on the performance of spatial analysis in a cyberinfrastructure environment — A case study with raster zonal analysis, *Computers & Geosciences*, 2016, vol. 87, pp. 11–21.
9. **Agosto E.** Vector-raster server-side analysis: a PostGIS benchmark, *Applied Geomatics*, 2013, vol. 5, no. 2, pp. 177–184.
10. **Ruzicka Z., Muzalevskiy K. V.** Web-GIS application for temperature and moisture monitoring of the arctic tundra soils based on SMOS data, *Regional problems of Earth Remote Sensing: Proceedings of the International Conference. Krasnoyarsk, Siberian Federal University*, 2014, pp. 287–290.
11. **Ruzicka Z., Muzalevskiy K. V.** Database approach to develop and validate the Soil Moisture and temperature retrieval algorithm using SMOS radiometric data, *2015 International Siberian Conference on Control and Communications (SIBCON)*. Proceedings, Omsk, Omsk State Technical University, pp. 1–5.
12. **Mironov V. L., Muzalevskiy K. V., Ruzicka Z.** Retrieving Profile Temperatures in a Frozen Topsoil Near the TFS, Alaska, Based on SMOS Brightness Temperatures at the 1.4 GHz Frequency, *IEEE Transactions on Geoscience and Remote Sensing*, 2016, vol. PP, is. 99, no. 7557037.
13. **Muzalevskiy K. V., Ruzicka Z.** Retrieving Soil Temperature at a Test Site on the Yamal Peninsula Based on the SMOS Brightness Temperature Observations, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2016, vol. 9, is. 6, pp. 2468–2477.
14. **Muzalevskiy K. V., Mikhailov M. I., Mironov V. L., Ruzicka Z.** Retrieving Soil Moisture and temperature using SMOS observations at a test site in the Yamal Peninsular, *2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, IEEE, 2016, pp. 4932–4935.
15. **Muzalevskiy K. V., Ruzicka Z., Kosolapova L. G., Mironov V. L.** Temperature dependence of SMOS/MIRAS, GCOM-W1/AMSR2 brightness temperature and ALOS/PALSAR radar backscattering at arctic test sites, *Progress in Electromagnetic Research Symposium (PIERS), IEEE*, 2016, pp. 3578–3582.
16. **Geoportal** of Institute of Computational Modelling, Siberian Branch of Russian Academy of Sciences. URL: <http://gis.krasn.ru/> (accessed 04.12.2016).

УДК 004.91

И. С. Печенко, науч. сотр., reizennn@mail.ru,
А. Б. Петров, д-р техн. наук, проф., зав. каф., petrov@mirea.ru,
Московский технологический университет

Модели данных для архитектурных спецификаций систем на кристалле

Сложность проектирования современных систем на кристалле постоянно растет, что влечет за собой рост важности анализа и верификации таких систем. Возникает потребность в подходе, который бы упростил переход от архитектурных спецификаций к высокоуровневым моделям системы, используемым для верификации. Работа представляет наиболее подходящие модели данных для создания спецификаций в рамках такого подхода, а также общую схему процесса работы со спецификациями. В большинстве случаев спецификацию системы на кристалле можно условно разделить на две части: структурную и поведенческую, поэтому модель данных и описание подхода также разделены на две части.

Ключевые слова: система на кристалле, архитектурная спецификация, верификация, автоматизация, модель данных, таблицы, диаграммы, структурные спецификации, поведенческие спецификации

Введение

Непрерывный рост сложности современных систем на кристалле (СНК) приводит к тому, что важнейшей частью процесса их проектирования становится верификация системы, входящих в нее IP-блоков и процессов их взаимодействия. Причем чем раньше начинается верификация системы, тем раньше можно будет выявить и исправить ошибки в системе и тем дешевле выявление этих ошибок обойдется разработчику системы [1, 2].

В традиционном процессе проектирования верификация системы начиналась только с момента готовности RTL-кода, так как до этого в распоряжении команды верификации не было достаточно формализованного представления системы, которое можно было бы подвергнуть автоматическим методам анализа. В последние десятилетия было разработано множество языков высокоуровневого моделирования [3, 4], и с их помощью стало возможным начинать верификацию системы намного раньше, как только закончена работа над архитектурной спецификацией. Существуют работы, посвященные созданию архитектурных спецификаций на формальных языках [5–7], однако в большинстве процессов проектирования такой подход не используется, так как:

— на создание архитектурных спецификаций на формальных языках требуется значительно больше времени и усилий;

— возникают сложности при использовании таких спецификаций на дальнейших этапах проектирования системы.

По этим причинам чаще всего используется следующий подход:

- архитектурные спецификации создают в виде текста на естественных языках с небольшими добавлениями в виде таблиц, схем, диаграмм и других формализованных данных;
- создание высокоуровневых моделей системы для ее верификации выполняется отдельной командой валидаторов, которые используют для этого готовые архитектурные спецификации.

К сожалению, такой переход от архитектурных спецификаций к высокоуровневым моделям системы сопряжен со значительными трудностями и требует больших временных затрат, что сильно затрудняет раннюю верификацию архитектуры системы. Эти трудности возникают в связи со свойственными написанным на естественных языках спецификациям недостатками, такими как неполнота, неточность, двусмысленность, а также в связи с часто присущей архитектурным спецификациям проблемой фрагментированности. Кроме того, также сильно затрудняет такой переход невозможность автоматической трансляции текстов на естественных языках, из которых состоят спецификации, на формальные языки.

В условиях постоянного повышения сложности проектируемых систем и диктуемого рынком сокращения времени, отведенного на их проектирование, возникает потребность в методике, соче-

тающей преимущества описанных подходов и нивелирующей их недостатки. Эта методика должна позволять:

- 1) повысить качество спецификаций, не требуя дополнительных усилий их создателей;
- 2) автоматизировать процесс переиспользования спецификаций;
- 3) автоматизировать процесс трансляции спецификаций на формальные языки.

Для оптимизации процесса создания архитектурных спецификаций систем на кристалле необходимо рассмотреть типичный состав этих спецификаций. Такие спецификации можно условно разделить на две части: структурную и поведенческую. Структурная часть спецификации описывает набор компонентов системы, их предполагаемое расположение на кристалле, специфику связей между этими компонентами (набор сообщений, которыми они обмениваются, интерфейсы компонентов), а также структуру компонентов, набор регистров в них и т. д. Поведенческая часть спецификации описывает алгоритмы поведения системы в различных ситуациях, а также алгоритмы взаимодействия компонентов системы. В качестве примеров поведенческих спецификаций можно привести протоколы перезагрузки системы, алгоритмы управления питанием в системе, протоколы безопасности и т. д.

На основе описания различных форматов для создания архитектурных спецификаций вычислительных систем, приведенного в работе [8], и применения к этим форматам выбранных критериев, можно заключить, что наиболее подходящими форматами данных для создания спецификаций являются:

— диаграммы с заданным синтаксисом, к которым могут относиться широко используемые в практике проектирования вычислительных систем диаграммы UML [9] (в число которых входит целый ряд видов диаграмм: диаграммы компонентов, диаграммы деятельности, диаграммы состояний, диаграммы коммуникации и другие), а также нотации BPMN [10] (Business Process Model Notation); такие диаграммы лучше всего подходят для описания поведенческих аспектов системы;

— таблицы, для которых также могут быть определены синтаксис и семантика [11], таблицы лучше всего подходят для описания структурных аспектов системы.

1. Процесс работы со спецификациями

Принимая во внимание недостатки, зачастую присущие архитектурным спецификациям (неполнота, неточность, двусмысленность, а также фрагментированность), можно отметить две важных составных части процесса создания спецификаций СнК. Он должен включать в себя автоматическую проверку корректности созданных спецификаций, решающую проблемы неполноты, неточности и

двусмысленности, а также создание связей между фрагментами спецификации, что решает проблему фрагментированности спецификации.

Важно помнить о втором требовании к исследуемому подходу. Так как зачастую для новых проектов, в особенности для СнК, значительная часть спецификаций создается путем переиспользования спецификаций прежних проектов [12], важным шагом процесса работы с архитектурными спецификациями является трансляция спецификаций из наиболее распространенных форматов данных в необходимые модели данных.

Наконец, последним из основных шагов процесса работы со спецификациями, согласно третьей поставленной задаче, должна стать трансляция спецификаций на формальные языки.

Опираясь на процесс проектирования систем на кристалле в целом, нетрудно представить, как в общих чертах будет выглядеть процесс работы со спецификациями.

1. Создание спецификаций на основе разработанных моделей данных либо с нуля, либо на основе существующих спецификаций из прошлых проектов. В последнем случае применяется функционал по трансляции данных из существующих спецификаций в разработанные модели данных.

2. Создание связей между фрагментами созданной спецификации.

3. Проверка корректности созданной спецификации.

4. Трансляция данных из созданной спецификации в различные формальные модели; дальнейшее использование этих моделей для верификации архитектуры и на следующих стадиях проектирования системы на кристалле.

Схема предлагаемого процесса работы со спецификациями представлена на рис. 1.

Остановимся подробнее на каждом шаге описанного выше процесса. Для трансляции данных необ-

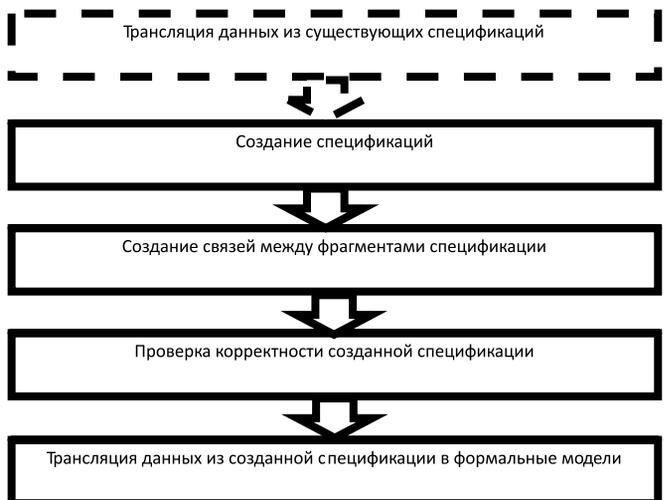


Рис. 1. Схема процесса работы со спецификациями

ходимо выбрать основные языки источников данных. Чаще всего в таких источниках используют рассмотренные в работе [8] форматы: естественные языки, диаграммы, таблицы и формальные языки. Для каждого из этих форматов необходимо подобрать наиболее подходящие для рассмотренных задач алгоритмы перевода в используемые модели данных.

Создание связей между фрагментами спецификаций может происходить несколькими способами. Первый и самый простой из них предполагает непосредственное создание каждой ссылки между двумя элементами спецификации с помощью тех же инструментов, которые используются для заполнения данных спецификаций. Второй способ предполагает создание классов ссылок между элементами спецификации по определенному признаку. В качестве примера можно привести случай, когда начальные элементы всех диаграмм связываются с первыми записями таблиц с определенным названием. При таком подходе создание классов ссылок и создание объектов-ссылок должно происходить в разное время.

Процесс проверки корректности спецификаций может быть разделен на три этапа: базовую проверку синтаксиса модели; проверку целостности и непротиворечивости данных; формальную верификацию модели.

Трансляция же данных из созданных спецификаций в формальные модели может выполняться с использованием тех же алгоритмов, которые применялись на первом шаге. При этом необходимо учитывать особенности используемых моделей данных.

Рассмотрим теперь особенности процесса работы со спецификациями в применении к спецификациям в формах диаграмм и таблиц.

2. Создание и использование поведенческих спецификаций

Первым шагом разработки процесса работы с поведенческими спецификациями должен стать выбор моделей данных для их представления.

Для поведенческих спецификаций модель данных должна основываться на диаграммах деятельности. В современных процессах проектирования ВС используют в основном два типа таких диаграмм: UML-диаграммы деятельности и нотации BPMN. Соответственно, модель данных для поведенческих спецификаций будет строиться на их основе.

Большая часть примитивов, необходимых для построения модели данных, имеется в диаграммах UML: действия; решения; агенты; примитивы начала и конца процесса; примитивы потока управления и передачи сообщения.

Единственное дополнение к этому списку объясняется важным свойством многих системных протоколов — возможностью представления их специ-

фикации на различных уровнях абстракции. Для этого необходимо добавить в модель данных поддержку иерархичного представления процессов, для чего можно использовать, в частности, подпроцессы и примитивы перехода по ссылкам, доступные в нотациях BPMN.

Рассмотрим подробнее шаги процесса работы со спецификациями в применении к поведенческим спецификациям в форме диаграмм.

Трансляция данных из существующих спецификаций в описанный формат диаграмм может быть осуществлена либо из таких же диаграмм последовательности операций UML и BPMN, либо из формальных языков, либо из текстового описания сценариев поведения системы. Трансляция из таких же диаграмм последовательности операций тривиальна и состоит только в приведении диаграмм к нужному виду, механизм трансляции из формальных языков необходимо разрабатывать отдельно для каждого языка, а на механизме трансляции из текстового описания сценариев поведения системы остановимся подробнее. Существующие работы, связанные с анализом и использованием текстовой спецификации, в большинстве случаев нацелены на улучшение процесса разработки программного обеспечения, однако их результаты в полной мере применимы и к процессу разработки спецификации СнК. Ряд работ предлагает способы экстракции UML-диаграмм из текстовой спецификации [13—15]. Общим для этих работ является схема процесса. Исходное текстовое описание сначала структурируется, что обеспечивает возможность последующей машинной обработки. За счет такого структурирования достигается соответствие текста некой формализованной структуре, которую затем нетрудно перевести в диаграмму UML. Такое структурирование достигается двумя основными путями: лингвистическим анализом с последующим устранением выявленных неточностей и дефектов в текстовых спецификациях и написанием спецификаций по заранее установленному структурированному шаблону.

Следующий этап процесса создания спецификаций — создание связей между ними. Оно может происходить тремя путями:

- записью ссылок на целевые объекты спецификации в свойствах элементов диаграммы;
- добавлением в диаграммы примитивов, служащих для определения ссылок между элементами диаграммы; в нотациях BPMN существует примитив "ассоциация", добавленный специально для таких целей. Минусом данного подхода является то, что такие связи можно установить только в рамках одной диаграммы;
- созданием специального документа, где будут храниться описания классов ссылок между элементами спецификации.

Все три способа задания ссылок между спецификациями в форме диаграмм могут быть использованы в рамках исследуемого процесса, и выбор того или иного способа зависит от решаемой задачи.

Что касается проверки корректности спецификаций в форме диаграмм, можно предложить два основных уровня такой проверки: простую проверку синтаксиса диаграммы и верификацию формальной модели, созданной на основе диаграммы. Простая проверка синтаксиса тривиальна, а верификацию формальных моделей, созданных на основе диаграммы, лучше всего осуществлять с помощью систем, специально созданных для верификации исполняемых моделей, так называемых систем проверки моделей.

Наконец, последний шаг процесса работы со спецификациями — их трансляция на формальные языки. Он необходим как для решения задачи формальной верификации протоколов, так и для автоматизации использования спецификаций протоколов на дальнейших стадиях проектирования. Первым шагом трансляции диаграмм последовательности операций в формальные модели должно являться приведение текста внутри элементов диаграммы к заданному виду.

После этого необходимо определить целевую формальную модель для трансляции. Наиболее подходящим вариантом такой модели для диаграммы действия является система состояний и переходов. Определим такую систему S через набор переменных состояния, набор начальных условий и набор правил перехода, представляющих собой пары "условие—действие", где условие — логическое выражение над переменными состояния, а действие — изменение значений переменных состояния.

Набор переменных состояния S представляет собой набор переменных диаграммы вместе со списком активных сообщений Q , с каждым элементом которого ассоциирован статус сообщения, и с подмножеством L всех ребер типа "последовательность", отображающим множество всех активных на данный момент ребер.

Начальное состояние системы характеризуется тем, что список Q и подмножество L пусты, а значения всех переменных диаграммы равны их начальным значениям.

Для каждого действия t определяем правило перехода τ системы S .

Кроме того, система S имеет дополнительные правила перехода, представляющие собой изменение статуса сообщений.

3. Создание и использование структурных спецификаций

Базовым форматом для модели данных основной части структурных спецификаций СнК были выбраны таблицы. Однако таблицы произвольного вида не удовлетворяют критерию возможности ав-

томатической проверки корректности табличных данных.

В литературе описано несколько подходов к проверке корректности таблиц произвольного вида [16—18], и все они являются полуавтоматическими. Дело в том, что формальный анализ табличных данных при отсутствии информации о структуре таблицы невозможен в большинстве случаев, поэтому все методы валидации табличных данных, не полагающиеся на знания структуры таблиц и предметной области, не полностью автоматизированы и требуют непосредственного участия пользователя. Наконец, выполнение еще одного критерия — нахождение возможности автоматической трансляции данных из таблиц произвольного вида — наиболее сложная задача при построении модели данных, основанной на таких таблицах.

Второй возможный путь — использование в качестве модели данных таблиц с предопределенной структурой. Такой подход к получению корректных табличных данных [11, 19—23] называется объектно-ориентированным. Этот подход предполагает соответствие синтаксиса и семантики таблиц заранее определенной модели. Процесс создания таких таблиц состоит из двух этапов: сначала создают их шаблоны, а затем отдельные экземпляры, соответствующие этим шаблонам. С помощью построения шаблонов с определенными свойствами такие таблицы можно реализовать по построению свободными от многих типовых для табличных данных ошибок, например, некорректных ссылок, типов и пробелов в записях.

Именно такой подход к созданию таблиц лучше всего удовлетворяет всем критериям оценки качества разрабатываемой модели данных. Опишем теперь структуру шаблона таблицы для разрабатываемой модели. Шаблон таблицы должен состоять из двух частей. Первая часть должна описывать внешний вид таблицы, а вторая часть — прочие характеристики таблицы: тип данных различных столбцов, ограничения на значения в них, их начальные значения, а также выражения для проверки корректности данных в таблице, описания возможных ссылок между данными и т. д. На рис. 2 представлен пример таблицы регистров. Материалы для примера взяты из описания открытой архитектуры OpenSPARC T2 [24].

Table name		Register definitions	
Comment		Summary of OpenSPARC T2 Privileged Registers	
Register Type	Register Address	Description	Access Type
TPC	00	Trap PC	RW
TNPC	01	Trap Next PC	RW
TSTATE	02	Trap State	RW
TT	03	Trap Type	RW
TICK	04	Tick	RW
TBA	05	Trap Base Address	RW
PSTATE	06	Process State	RW
TL	07	Trap Level	RW

Рис. 2. Пример таблицы регистров

Для данного примера текст шаблона таблицы в формате XML может выглядеть следующим образом:

```
<table_template name = "Register definitions">
  <header>
    <info type = "string">Table name</info>
    <info type = " string">Comment</info>
  </header>
  <body>
    <field type = " string" unique = "true">Register
    Type</field>
    <field type = " int">Register Address</field>
    <field type = " string">Description</field>
    <field type = " enum" values = "RW,RO">Access
    Type</field>
  </body>
</table_template>
```

Трансляция данных из существующих спецификаций в табличный формат возможна либо из другого табличного формата, либо из формальных языков. Если для каждого формального языка необходимо разработать свой подход для трансляции, то перевод таблицы в таблицу несложен, однако в большинстве случаев все же требует участия человека. Простейшим способом перевода данных из одного формата таблиц в другой является перевод на основе соответствия полей по имени либо номеру, когда для каждого поля целевой таблицы пользователь напрямую указывает соответствующее ему поле таблицы-источника.

Хотя объектно-ориентированный подход к разработке таблиц направлен на разработку корректных по построению таблиц, так как в шаблоне таблицы описаны базовые требования к данным в записях таблицы, некоторые данные в них все равно необходимо подвергать процессу валидации. Наиболее подходящим способом проведения такой валидации является помещение в шаблон таблицы списка выражений для автоматической проверки целостности и непротиворечивости данных в ней. Подобный вариант валидации табличных данных, использующий проверку заданных пользователем утверждений, описан в работе [25]. Однако этот механизм можно расширить, добавив возможность не только валидировать конкретные записи таблицы, но и добавлять требования к существованию или единственности определенных записей.

Каждое описание проверки корректности спецификации должно содержать следующие записи.

1. Логическое выражение для проверки, принимающее значение "true" в случае корректности спецификации.

2. Описание переменных, используемых в этом логическом выражении. Так как нашей целью является поддержка выражений, использующих данные из всех записей таблицы, нужно предусмотреть следующие возможности языка описания переменных.

2а. Выбор значения поля или комбинации значений нескольких полей из каждой записи таблицы или из каждой записи из определенного множества записей.

2б. Выбор множества записей таблицы, обладающих определенным признаком (например, всех записей, значение поля "name" в которых равно "register1").

2с. Выбор множества записей таблицы, состоящих в определенном отношении с записями, выбранными для другой переменной. В качестве примеров можно привести два случая: для переменной v2 нужно выбрать все записи в таблице, кроме записи, выбранной для переменной v1; для переменной v2 нужно выбрать запись в таблице, следующую за записью, выбранной для переменной v1.

Реализация такого языка описания проверок может быть различной, но крайне желательно, чтобы она поддерживала описанные возможности.

Наконец, последней составной частью процесса работы с табличными спецификациями является трансляция ее на различные формальные языки. Алгоритмы трансляции необходимо разрабатывать отдельно для каждого шаблона таблиц и для каждого целевого языка. В качестве примера можно предложить следующий алгоритм трансляции таблицы регистров на язык SystemRDL [26].

1. Создаем блок регистров для каждого уникального значения поля "block name" в таблице.

2. Для каждой записи в таблице создаем регистр с именем, соответствующим значению поля "register name" в блоке с именем, соответствующим значению поля "block name".

3. Задаем характеристики каждого регистра: поле размера заполняем значением поля "bit width"; поле характеристики доступа — значением поля "access type"; поле начального значения — значением поля "default value"; поле типа памяти — значением поля "storage type" и т. д.

Заключение

В данной работе описаны модели данных для архитектурных спецификаций систем на кристалле и подход к работе с ними, необходимый для решения проблемы позднего начала верификации систем и повышения качества архитектурных спецификаций. Для построения моделей данных спецификацию SnK условно делят на две части: структурную и поведенческую. Для каждой из этих частей используют свою модель данных: таблицы с заданными шаблонами для структурных спецификаций и диаграммы последовательности операций (UML-диаграммы деятельности или нотаций BPMN) для поведенческой спецификации.

Процесс работы со спецификациями включает в себя четыре стадии.

1. Создание спецификации либо с нуля, либо на основе существующих спецификаций, использовавшихся в предыдущих проектах.

2. Создание связей между фрагментами спецификации.
3. Проверка корректности спецификации.
4. Трансляция спецификации или ее фрагментов на формальные языки для использования при построении высокоуровневых моделей системы для верификации либо использования на дальнейших этапах проектирования системы.

Список литературы

1. **Lee Y., Kim N., Kim D., Lee D., In H.** Customer Requirements Elicitation based on Social Network Service // KSTT Transactions On Internet And Information Systems. October 2011. Vol. 5. N. 10. P. 1733—1750.
2. **Neetu Kumari S., Pillai A. S.** A survey on global requirements elicitation issues and proposed research framework // Software Engineering and Service Science (ICSESS). 4th IEEE International Conference. 2013. P. 554—557.
3. **Riccobene E., Scandurra P., Rosti A., Bocchio S.** A SoC design methodology involving a UML 2.0 profile for SystemC // Design, Automation and Test in Europe. 2005. P. 704—709.
4. **Pees S., Hoffmann A., Zivojnovic V., Meyr H.** LISA — machine description language for cycle-accurate models of programmable DSP architectures // In Proceedings of the 36th annual ACM/IEEE Design Automation Conference. June 1999. P. 933—938.
5. **Schliebusch O., Hoffmann A., Nohl A., Braun G., Meyr H.** Architecture implementation using the machine description language LISA // Design Automation Conference, 2002. Proceedings of ASP-DAC 2002. 7th Asia and South Pacific and the 15th International Conference on VLSI Design. 2002. P. 239—244.
6. **Hoffmann A., Nohl A., Braun G., Meyr H.** A survey on modeling issues using the machine description language LISA // Acoustics, Speech, and Signal Processing, 2001. Proceedings. IEEE International Conference. 2001. Vol. 2. P. 1137—1140.
7. **Wieferink A., Doerper M., Leupers R., Ascheid G., Meyr H., Kogel T., Braun G., Nohl A.** System level processor/communication co-exploration methodology for multi-processor system-on-chip platforms // Computers and Digital Techniques, IEEE Proceedings — January 2005. Vol. 152. No. 1. P. 3—11.
8. **Печенко И. С.** Способы представления спецификаций вычислительных систем: проблемы и возможности машинной обработки // Информационные технологии. 2016. Т. 22, № 9. С. 676—683.
9. **Rumbaugh J., Jacobson I., Booch G.** Unified modeling language reference manual. Boston, USA: Pearson Higher Education, 2004. 742 p.
10. **Описание** графических нотаций BPMN. URL: <http://www.bpmn.org/> (дата обращения: 01.01.2017).
11. **Abraham R., Erwig M., Kollmansberger S., Seifert E.** Visual specifications of correct spreadsheets // Visual Languages and Human-Centric Computing, 2005 IEEE Symposium. 2005. P. 189—196.
12. **Thompson J., Heimdahl M., Erickson D.** Structuring formal control systems specifications for reuse: Surviving hardware changes // Lfm2000: Fifth NASA Langley Formal Methods Workshop. National Aeronautics and Space Administration, Langley Research Center, 2000. Vol. 210100. P. 117.
13. **Gelhausen T., Tichy W.** Thematic Role Based Generation of UML Models from Real Word Requirements // In Proceedings of the 1st IEEE International Conference on Semantic Computing. 2007. P. 282—289.
14. **Sharma V., Sarkar S., Verma K., Panayappan A., Kass A.** Extracting High-level Functional Design from Software Requirements // In Proceedings of the 16th IEEE Asia-Pacific Software Engineering Conference. 2009. P. 35—42.
15. **Deeptimahanti D. K., Sanyal R.** An Innovative Approach for Generating Static UML Models from Natural Language Requirements // Advances in Software Engineering, Communication in computer and Information Science. 2009. Vol. 30. P. 147—163.
16. **Abraham R., Erwig M.** Goal-directed debugging of spreadsheets // Visual Languages and Human-Centric Computing, 2005 IEEE Symposium. 2005. P. 37—44.
17. **Abraham R., Erwig M.** GoalDebug: A spreadsheet debugger for end users // Proceedings of the 29th international conference on Software Engineering. 2007. P. 251—260.
18. **Fisher M., Rothermel G., Creelan T., Burnett M.** Scaling a Dataflow Testing Methodology to the Multi-paradigmWorld of Commercial Spreadsheets // Software Reliability Engineering, 2006. ISSRE'06. 17th International Symposium. 2006. P. 13—22.
19. **Mendes J.** Coupled evolution of model-driven spreadsheets // Software Engineering (ICSE), 2012 34th International Conference. 2012. P. 1616—1618.
20. **Cunha J., Fernandes J. P., Mendes J., Saraiva J.** MDSheet: A framework for model-driven spreadsheet engineering // Proceedings of the 34th International Conference on Software Engineering. 2012. P. 1395—1398.
21. **Engels G., Erwig M.** ClassSheets: automatic generation of spreadsheet applications from object-oriented specifications // Proceedings of the 20th IEEE/ACM international Conference on Automated software engineering. 2005. P. 124—133.
22. **Cunha J., Fernandes J. P., Mendes J., Pacheco H., Saraiva J.** Bidirectional Transformation of Model-Driven Spreadsheets // IC-MT. 2012. Vol. 12. P. 105—120.
23. **Stevens P.** Bidirectional model transformations in QVT: Semantic issues and open questions // Model Driven Engineering Languages and Systems. 2007. P. 1—15.
24. **Спецификация** архитектуры OpenSparc-T2. URL: <http://www.oracle.com/technetwork/systems/opensparc/opensparc-t2-page-1446157.html> (дата обращения: 01.07.2017).
25. **Burnett M., Cook C., Pendse O., Rothermel G., Summet J., Wallace C.** End-user software engineering with assertions in the spreadsheet paradigm // In Proceedings of the 25th International Conference on Software Engineering. May 2003. P. 93—103.
26. **Описание** языка SystemRDL. URL: <http://accellera.org/activities/working-groups/systemrdl> (дата обращения: 01.01.2017).

I. S. Pechenko, Researcher, **A. B. Petrov**, D. Sc., Professor, Head of the Chair, Moscow Technological University (MIREA), reizennn@mail.ru, petrov@mirea.ru

Data Models for SoC Specifications Creation and Usage

The complexity of systems on chips (SoCs) design significantly grows every year, and it leads to increasing the importance of SoC architecture verification process. Development of high-level modeling languages in the past decades allowed starting verification very early, when system architecture specification is ready. Despite this, there still is a gap between architecture specification creation and the start of system verification. The common way of transition from architecture specification creation to executable model design is to create the specification mostly using natural language and then manually create the executable model based on it. To close this gap, we need to create specifications that are free of typical architecture specifications drawbacks: ambiguity, incompleteness, fragmentation and easy to create and understand. This paper presents a data model for SoC specifications and specification creation and usage process that can help architects to solve this problem.

SoC architectural specification can be divided into two parts: structural specification and behavioral specification. The first one is a map of system components and a requirements specification for each of them. The second is a description of system behavior in different situations and protocols of IP-block intercommunication.

The basic specifications usage process should be designed to solve the problem of transition from specification to high-level model and take care of all potential specification drawbacks. Based on this, the process should be as follows:

1. Specification creation from scratch or from existing architecture.
2. Binding all parts of the specification together.
3. Data validation. This process can be divided in three parts: syntax check, consistency check and formal verification.
4. Automatic specification translation to high-level formal model.

Very important step of specification creation process design is data model definition. Data model should negate the drawbacks of traditional specifications and should be easy enough to be used for specification creation and to understand for any specification user. The best data presentation format for behavioral specifications is behavior diagrams, and there are two behavior diagram languages that are used most often for system specification design: UML activity diagrams and BPMN notations. One of the most common formats for structural specifications is spreadsheet format. To support data validation and automatic transition to formal models we chose object-oriented approach to spreadsheet creation, when at first a class of spreadsheets is created and then individual spreadsheets are instantiated from this class. All necessary properties of a spreadsheet class can be written in an XML-document attached to it.

Keywords: system on a chip (SoC), architectural specification, verification, automation, data model, spreadsheets, diagrams, structural specifications, behavioral specifications

References

1. Lee Y., Kim N., Kim D., Lee D., In H. Customer Requirements Elicitation based on Social Network Service, *KSTT Transactions On Internet And Information Systems*, October 2011, vol. 5, no. 10, pp. 1733–1750.
2. Neetu Kumari S., Pillai A. S. A survey on global requirements elicitation issues and proposed research framework, *Software Engineering and Service Science (ICSESS)*. 4th IEEE International Conference, 2013, pp. 554–557.
3. Riccobene E., Scandurra P., Rosti A., Bocchio S. A SoC design methodology involving a UML 2.0 profile for SystemC, *Design, Automation and Test in Europe*, 2005, pp. 704–709.
4. Pees S., Hoffmann A., Zivojnovic V., Meyr H. LISA — machine description language for cycle-accurate models of programmable DSP architectures, *In Proceedings of the 36th annual ACM/IEEE Design Automation Conference*, June 1999, pp. 933–938.
5. Schliebusch O., Hoffmann A., Nohl A., Braun G., Meyr H. Architecture implementation using the machine description language LISA, *Design Automation Conference, 2002. Proceedings of ASP-DAC 2002. 7th Asia and South Pacific and the 15th International Conference on VLSI Design*, 2002, pp. 239–244.
6. Hoffmann A., Nohl A., Braun G., Meyr H. A survey on modeling issues using the machine description language LISA, *Acoustics, Speech, and Signal Processing, 2001. Proceedings. IEEE International Conference*, 2001, vol. 2, pp. 1137–1140.
7. Wiefierink A., Doerper M., Leupers R., Ascheid G., Meyr H., Kogel T., Braun G., Nohl A. System level processor/communication co-exploration methodology for multi processor system-on-chip platforms, *Computers and Digital Techniques, IEEE Proceedings*, January 2005, vol. 152, no. 1, pp. 3–11.
8. Pechenko I. Specification representation forms: issues and ways for automatic processing. *Information technologies*, 2016, vol. 22, N. 9, pp. 676–683.
9. Rumbaugh J., Jacobson I., Booch G. *Unified modeling language reference manual*, Boston, USA, Pearson Higher Education, 2004, 742 p.
10. BPMN graphical notations description. URL: <http://www.bpmn.org/> (date of appeal: 01.01.2017).
11. Abraham R., Erwig M., Kollmansberger S., Seifert E. Visual specifications of correct spreadsheets, *Visual Languages and Human-Centric Computing, 2005 IEEE Symposium*. 2005. pp. 189–196.
12. Thompson J., Heimdahl M., Erickson D. Structuring formal control systems specifications for reuse: Surviving hardware changes, *Lfm2000: Fifth NASA Langley Formal Methods Workshop. National Aeronautics and Space Administration, Langley Research Center, 2000*. vol. 210100. pp. 117.
13. Gelhausen T., Tichy W. Thematic Role Based Generation of UML Models from Real Word Requirements, *In Proceedings of the 1st IEEE International Conference on Semantic Computing*, 2007, pp. 282–289.
14. Sharma V., Sarkar S., Verma K., Panayappan A., Kass A. Extracting High-level Functional Design from Software Requirements, *In Proceedings of the 16th IEEE Asia-Pacific Software Engineering Conference*, 2009. pp. 35–42.
15. Deeptimahanti D. K., Sanyal R. An Innovative Approach for Generating Static UML Models from Natural Language Requirements, *Advances in Software Engineering, Communication in computer and Information Science*, 2009, vol. 30, pp. 147–163.
16. Abraham R., Erwig M. Goal-directed debugging of spreadsheets, *Visual Languages and Human-Centric Computing, 2005 IEEE Symposium*, 2005, pp. 37–44.
17. Abraham R., Erwig M. GoalDebug: A spreadsheet debugger for end users, *Proceedings of the 29th international conference on Software Engineering*, 2007, pp. 251–260.
18. Fisher M., Rothermel G., Creelan T., Burnett M. Scaling a Dataflow Testing Methodology to the Multi paradigm World of Commercial Spreadsheets, *Software Reliability Engineering, 2006. ISSRE'06. 17th International Symposium*, 2006, pp. 13–22.
19. Mendes J. Coupled evolution of model-driven spreadsheets, *Software Engineering (ICSE), 2012 34th International Conference*, 2012, pp. 1616–1618.
20. Cunha J., Fernandes J. P., Mendes J., Saraiva J. MDSheet: A framework for model-driven spreadsheet engineering, *Proceedings of the 34th International Conference on Software Engineering*, 2012, pp. 1395–1398.
21. Engels G., Erwig M. ClassSheets: automatic generation of spreadsheet applications from object-oriented specifications, *Proceedings of the 20th IEEE/ACM international Conference on Automated Software Engineering*, 2005, pp. 124–133.
22. Cunha J., Fernandes J. P., Mendes J., Pacheco H., Saraiva J. Bidirectional Transformation of Model-Driven Spreadsheets, *ICMT*, 2012, vol. 12, pp. 105–120.
23. Stevens P. Bidirectional model transformations in QVT: Semantic issues and open questions, *Model Driven Engineering Languages and Systems*, 2007, pp. 1–15.
24. OpenSparc-T2 architecture specification. URL: <http://www.oracle.com/technetwork/systems/opensparc/opensparc-t2-page-1446157.html> (date of appeal: 01.07.2017).
25. Burnett M., Cook C., Pendse O., Rothermel G., Summet J., Wallace C. End-user software engineering with assertions in the spreadsheet paradigm, *In Proceedings of the 25th international conference on Software Engineering*, May 2003, pp. 93–103.
26. SystemRDL language description. URL: <http://accellera.org/activities/working-groups/systemrdl> (date of appeal: 01.01.2017).

НЕЙРОСЕТЕВЫЕ ТЕХНОЛОГИИ NEUROTECHNOLOGIES

УДК 004.3

П. Ш. Гейдаров, канд. техн. наук, доц., e-mail: plbaku2010@gmail.com
Институт системного управления НАН Азербайджана, г. Баку

Нейронные сети прямого распространения с вычисляемыми параметрами

Описывается процедура создания нейронных сетей прямого распространения с определяемой архитектурой и вычисляемыми параметрами сети, реализующая алгоритм распознавания на основе метрических методов распознавания с использованием метрических характеристик близости. Приведены примеры вычисления таблиц весов связей нейронной сети без применения классических алгоритмов обучения нейронной сети. Описаны возможности минимизации числа нейронов и связей для полученных сетей. Описан алгоритм обучения, реализующий отбор минимального набора эталонов из обучающей выборки.

Ключевые слова: модели нейронных сетей, обучение нейронных сетей, линейные нейронные сети, метод ближайшего соседа, метод построения эталона, метрические методы распознавания, функция активации

Введение

В настоящее время по-прежнему остается актуальным вопрос создания простых и понятных архитектур нейронных сетей [1–4]. В данной работе рассматриваются возможности создания архитектуры нейронной сети, параметры которой определяются или вычисляются аналитически на основе метрических методов распознавания [5]. Простота и математическая точность алгоритмов метрических методов распознавания позволяет создать прозрачную нейронную сеть, где число нейронов, слоев, связей, а также пороговых значений определяется предварительно в зависимости от начальных условий задачи, таких как: число образов, эталонов, признаков, что выгодно отличает эти сети от классических схем создания нейронных сетей прямого распространения. Здесь эталоны — это выделенные объекты (образцы) образа изображения, выбранные из существующей выборки объектов образа, либо образцы, полученные путем объединения параметров разных объектов образа в один образцовый эталон [5]. Возможности предлагаемых сетей позволяют также аналитически определять весовые значения связей без применения классических алгоритмов обучения [6–9] на основе применяемых метрических характеристик близости.

Метод построения эталона и нейронная сеть

Рассмотрим возможность создания нейронной сети на основе алгоритма распознавания эталонов применительно к задаче распознавания кривых. Пусть в качестве эталонов имеется по одному выделенному эталону кривой для каждого образа. Если число эталонных кривых одного образа больше

единицы, то в этом случае будем определять средний эталон (метод построения эталона [5]), параметры которого вычисляются по формуле

$$y_i = \frac{1}{K} \sum_{k=0}^K y_{ik}, \quad (1)$$

где y_i — i -й параметр среднего эталона; y_{ik} — i -й параметр k -го эталона; K — число выбранных эталонов одного образа.

На рис. 1 приведены две эталонные кривые $etal_1$ и $etal_2$. При этом среднеквадратичная близость тестируемой кривой к эталонной будет вычисляться по формуле

$$S = \frac{1}{M_1} \sum_{i=0}^{M_1} (y_{etal1_i} - y_{test_i})^2 - \frac{1}{M_2} \sum_{i=0}^{M_2} (y_{etal2_i} - y_{test_i})^2, \quad (2)$$

где S — коэффициент, определяющий близость кривых; y_{etal1_i} , y_{etal2_i} — значения эталонных кривых i -го отсчета по оси x ; y_{test_i} — значения тестируемой кривой i -го отсчета по оси x ; M_1 , M_2 — число дискретных отсчетов для каждой кривой по оси x . В случае если $M_1 = M_2$, то коэффициент S определяется выражением

$$S = \sum_{i=0}^M (y_{etal1_i} - y_{test_i})^2 - \sum_{i=0}^M (y_{etal2_i} - y_{test_i})^2, \quad (3)$$

где M — число дискретных отсчетов для каждой кривой.

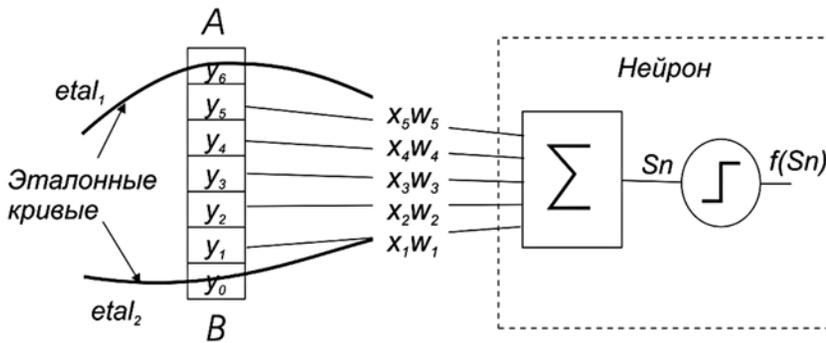


Рис. 1. Схема получения одного нейрона

На рис. 1 приведен пример столбца изображения AB , разбитый на участки, при этом между двумя эталонными кривыми $etal_1$ и $etal_2$ имеются пять прямоугольных участков. Если тестируемая кривая $test$ на этом отсчете x проходит между двумя эталонными кривыми, то соответственно она пройдет через один из этих участков. В этом случае согласно выражению (3) для этого участка i можно вычислить значение веса w_i :

$$w_i = (y_{etal1A} - y_{test_i})^2 - (y_{etal2B} - y_{test_i})^2, \quad (4)$$

где y_{etal1A} и y_{etal1B} — значения кривых $etal_1$ и $etal_2$ для этого столбца.

Если значение $y_{etal1A} = 6$, $y_{etal2B} = 0$ (рис. 1), то для разных значений y_{test_i} получим веса w_i для всех участков между двумя кривыми в столбце AB :

$$\begin{aligned} w_1 &= (6 - 1)^2 - (0 - 1)^2 = 25 - 1 = 24, \\ w_2 &= (6 - 2)^2 - (0 - 2)^2 = 16 - 4 = 12, \\ w_3 &= (6 - 3)^2 - (0 - 3)^2 = 9 - 9 = 0, \\ w_4 &= (6 - 4)^2 - (0 - 4)^2 = 4 - 16 = -12, \\ w_5 &= (6 - 5)^2 - (0 - 5)^2 = 1 - 25 = -24. \end{aligned} \quad (5)$$

Полученные значения w_i , а также соответствующие значения x_i подаются на нейрон. При этом $x_i = 1$, если тестируемая кривая проходит через данный участок, и $x_i = 0$, если кривая не проходит

через данный участок. В итоге значение нейрона для одного столбца AB вычисляется по формуле

$$Sn = \sum_{i=0}^K x_i w_i, \quad (6)$$

где x_i — значение x для i -й строки; w_i — вес для i -й строки; K — число участков в столбце AB ; Sn — значение нейрона для столбца AB .

Аналогично можно определить значение Sn для всех остальных столбцов изображения. В этом случае общее значение нейрона

$$Sn = \sum_{j=0}^M \sum_{i=0}^K x_{ij} w_{ij}, \quad (7)$$

где x_{ij} — значение x для i -й строки и j -го столбца; w_{ij} — вес для i -й строки и j -го столбца; M, K — число столбцов и строк изображения.

При этом значение w_{ij} будет определяться формулой

$$w_{ij} = (y_{etal1_j} - y_{test_{ij}})^2 - (y_{etal2_j} - y_{test_{ij}})^2, \quad (8)$$

где y_{etal1_j}, y_{etal2_j} — значения эталонных кривых j -го отсчета; $y_{test_{ij}}$ — значение отсчета i -й строки и j -го столбца.

В отличие от классических подходов [6–9] определения весовых и пороговых значений, для которых требуются длительные и сложные алгоритмы обучения нейронных сетей, выполняющие постепенную настройку значения веса связи нейрона с многочисленными итерациями обучения, выражение (8) позволяет получить значения веса связи нейрона первого слоя за один шаг. Такой подход значительно ускоряет процесс создания нейронной сети прямого распространения.

На рис. 2 приведена таблица весов для диапазона изображения кривых, разбитых на 35 участков. Можно видеть, что ячейки, находящиеся ближе к эталонной кривой 2, имеют положительные значения, и наоборот, ячейки, находящиеся ближе к кривой 1, имеют отрицательные значения, т. е. выполняется условие:

$$\begin{aligned} f(Sn) &= 1, \text{ если } Sn < 0; \\ f(Sn) &= 0, \text{ если } Sn > 0, \end{aligned} \quad (9)$$

где $f(Sn)$ — пороговая функция активации нейрона.

Таким образом, получаем нейрон, значение которого определяет близость к одному из образов кривых.

Приведенный пример соответствует случаю, когда имеются два эталонных образа кривых, и для этого достаточно было использование одного нейрона (см. рис. 1). Если число образов больше

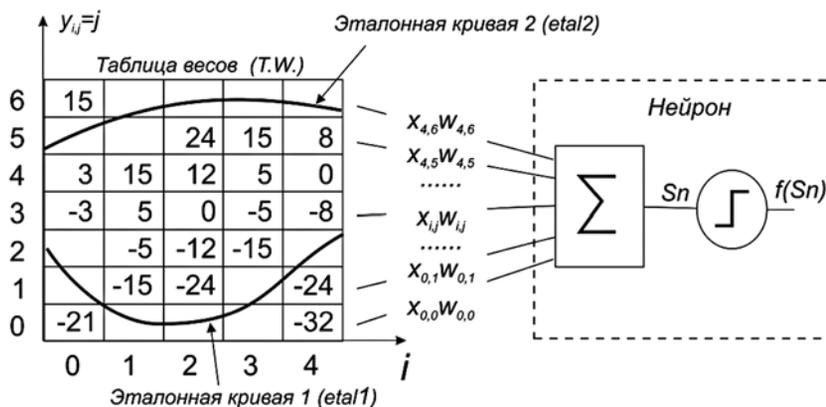


Рис. 2. Таблица весов (Т. В.)

двух, то в этом случае для каждой пары эталонных кривых создается свой нейрон. На рис. 3, а приведен пример первого слоя нейронной сети для распознавания трех образов кривых.

При этом если число распознаваемых образов равно N , то общее число нейронов n первого слоя будет равно

$$n = N(N - 1). \quad (10)$$

Для получения второго слоя нейронов будем исходить из того, что если тестируемая кривая будет близка к какой-то эталонной кривой $etal_n$, то все попарные сравнения этой кривой $etal_n$ с другими эталонами будут также указывать на близость распознаваемой кривой к эталонной кривой $etal_n$. При этом в первом нейроне второго слоя будут суммироваться выходы нейронов первого слоя, которые связаны с первым эталонным образом кривой (на рис. 3, а первые два нейрона) и т. д. В этом случае пороговая функция активации $f(Sn)$ для нейронов второго слоя определится условиями:

$$\begin{aligned} f(Sn_i) &= 1, \text{ если } Sn_i \geq \alpha(N - 1), \\ f(Sn_i) &= 0, \text{ если } Sn_i < \alpha(N - 1), \end{aligned} \quad (11)$$

где $\alpha(N - 1)$ — пороговое значение нейрона; Sn_i — значение i -го нейрона второго слоя; α — значение веса w_i . В простейшем случае $\alpha = 1$.

На рис. 3, а в первом слое сети можно видеть нейроны, которые выполняют идентичные функции, например, нейроны 1—2, 2—1. В целях минимизации числа нейронов в первом слое, нейроны, которые повторяют равнозначные действия, исключаются из схемы сети. При этом общее число нейронов n первого слоя уменьшится в 2 раза, и для задачи распознавания N образов будет определяться формулой

$$n = C_N^2 = \frac{N!}{2!(N-2)!} = \frac{N(N-1)}{2}. \quad (12)$$

На рис. 4 приведена схема нейронной сети для задачи распознавания пяти образов кривых. На каждый нейрон первого слоя подаются все значения x_{ij} и значения весов w_{ij} из таблицы весов (см. рис. 2). Таблицы весов для каждого нейрона различны и вычисляется по выражению (8). На рис. 4, в отличие от схемы на рис. 3, выходы исключенных нейронов будут заменяться оставшимися и соответствующими им нейронами, значения которых будут инвертироваться и подаваться на входы нужного нейрона второго слоя (рис. 4), что позволяет сохранить целостность выполнения всех нейронов второго слоя, функция активации которых будут вычисляться по выражению (11).

На схемах рис. 3, 4 число выходов нейронной сети равно числу образов эталонных кривых. При этом только один из выходов второго слоя будет равен единице, порядковый номер которого будет соответствовать номеру образа, например, для ва-

рианта с пятью образами на схеме рис. 4 код на выходе сети для первого образа будет равен 10 000, для второго 01 000 и т.д.

На рис. 5 приведена общая схема нейронной сети для задачи распознавания N образов. Схема явля-

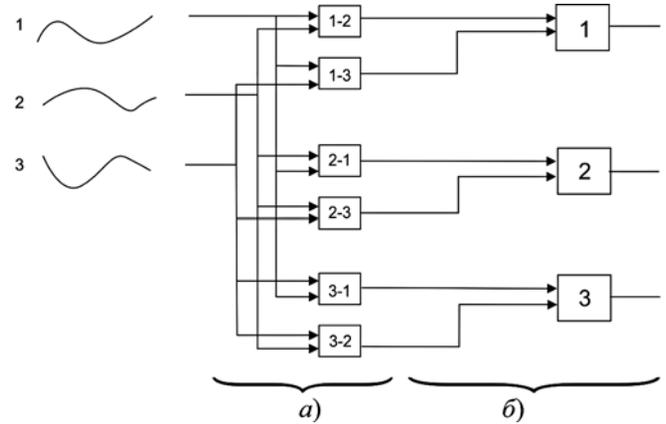


Рис. 3. Нейронная сеть для трех образов кривых: а — первый слой; б — второй слой

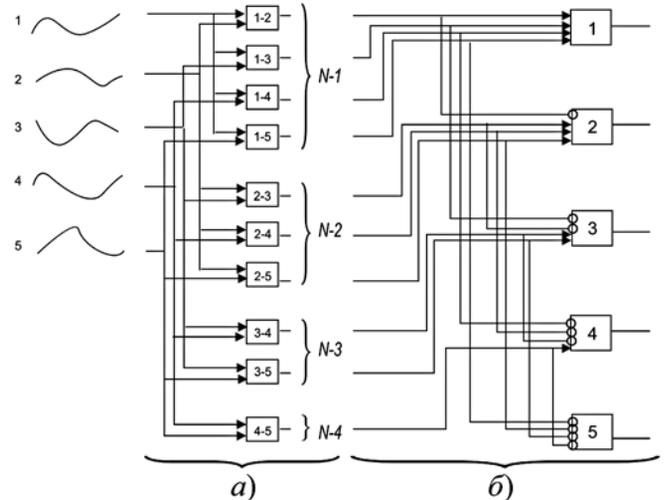


Рис. 4. Нейронная сеть для пяти образов кривых: а — первый слой; б — второй слой

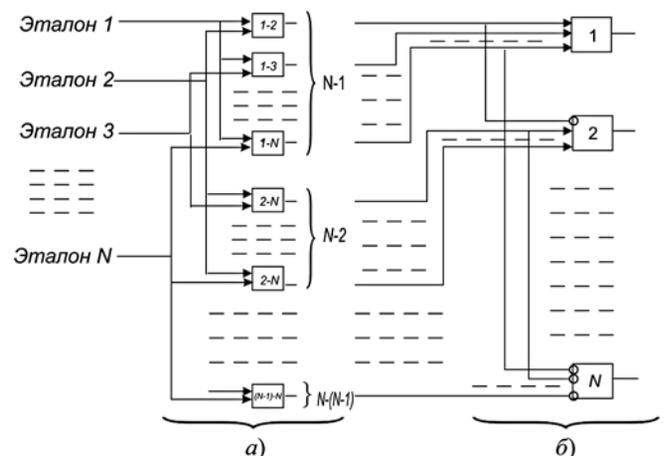


Рис. 5. Общая схема для распознавания N кривых: а — первый слой; б — второй слой

ется минимизированной по числу нейронов и связей между ними.

В целом логика работы сети, приведенной на рис. 5, выглядит следующим образом. Первый слой сети выполняет попарное сравнение всех эталонов, второй слой объединяет результаты выходов первого слоя, соответствующие одному эталону в одном нейроне. Активный выход i -го нейрона второго слоя означает соответствие распознаваемого объекта i -му эталону. В случае, если в качестве эталонов используются средние эталоны (1), то в этом случае данная логика нейронной сети соответствует алгоритму метода построения эталона [5].

Метод ближайшего соседа и нейронная сеть

Как было отмечено выше, описанные алгоритм и схемы (рис. 4, 5) были приведены для случая, когда для каждого распознаваемого образа имеется либо явно заданный эталон, либо из нескольких эталонов, соответствующих одному образу, создается один средний эталон согласно выражению (1). Если в процессе распознавания одновременно использовать несколько эталонов, то в этом случае на выходе второго слоя одному образу будет соответствовать несколько выходов K , равных по количеству числу эталонов этого образа. Объединение выходов эталонов, соответствующих одному образу, выполняется в нейроне третьего слоя (рис. 6). Таким образом, реализуется метод ближайшего соседа [5].

При этом число нейронов для третьего слоя будет равно числу образов, для которых число эталонов больше единицы, а функция активации для нейрона третьего слоя (σ) будет определяться согласно условиям

$$\begin{aligned} f(Sn_i) &= 1, \text{ если } Sn_i > 0; \\ f(Sn_i) &= 0, \text{ если } Sn_i = 0, \end{aligned} \quad (13)$$

где Sn_i — значение i -го нейрона третьего слоя. При описанном подходе каждый эталон одного образа представляется как эталон отдельного образа. Тем не менее, несмотря на работоспособность полученной схемы, в данной схеме может быть выполнена дополнительная минимизация числа нейронов в первом слое. Для дальнейшей минимизации этой сети необходимо исключить нейроны в первом слое, которые выполняют попарное сравнение эталонов, соответствующих одному и тому же образу. В этом случае число нейронов в первом слое будет определяться формулой

$$\begin{aligned} n &= n_1(N_{etal} - n_1) + n_2(N_{etal} - n_1 - n_2) + \dots + \\ &+ n_j \left(N_{etal} - \sum_{i=1}^j n_i \right) + \dots + n_{N-1} \left(N_{etal} - \sum_{i=1}^{N-1} n_i \right) = \\ &= \sum_{j=1}^{N-1} n_j \left(N_{etal} - \sum_{i=1}^j n_i \right), \end{aligned} \quad (14)$$

где n_j, n_i — число эталонов j -го и i -го образов; N_{etal} — общее число всех эталонов; N — число образов; n — число нейронов в первом слое. При этом число нейронов второго слоя будет равно общему числу эталонов N_{etal} , а функция активации нейрона второго слоя будет определяться условием

$$\begin{aligned} f(Sn_i) &= 1, \text{ если } Sn_i \geq \alpha(N_{etal} - n_j); \\ f(Sn_i) &= 0, \text{ если } Sn_i < \alpha(N_{etal} - n_j), \end{aligned} \quad (15)$$

где Sn_i — значение i -го нейрона второго слоя, $\alpha(N_{etal} - n_j)$ — пороговое значение нейрона, N_{etal} — общее число всех эталонов, n_j — число эталонов j -го образа, α — постоянная константа — значение веса w_j .

Способы представления объектов распознавания при подаче на нейронную сеть могут быть различными, например, однотонные изображения-символы могут быть представлены в виде кривых. При этом в зависимости от алгоритма преобразования итоговые кривые могут быть однослойными (рис. 7, а) или многослойными (рис. 7, б).

Для многослойных кривых при создании таблицы весов будет выполняться сравнение соответствующих слоев кривой с соответствующими слоями кривых другого образа. При этом если число слоев

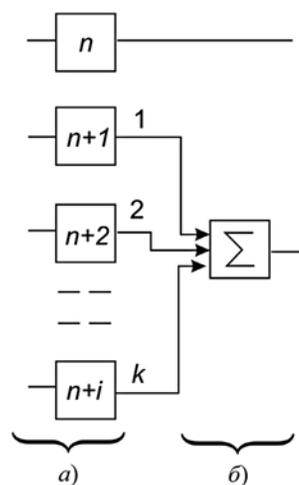


Рис. 6. Объединение эталонов одного образа в нейроне третьего слоя: а — второй слой; б — третий слой

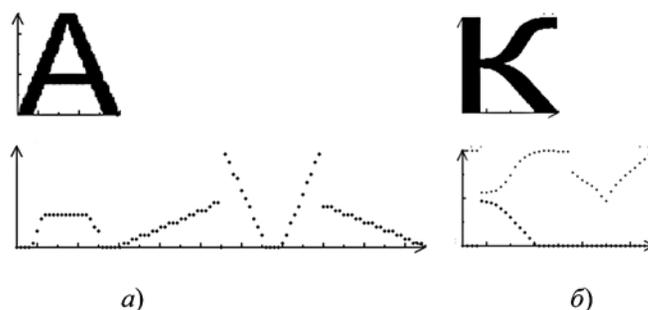


Рис. 7. Кривые преобразования для символов "А", "К": а — однослойная кривая; б — двухслойная кривая

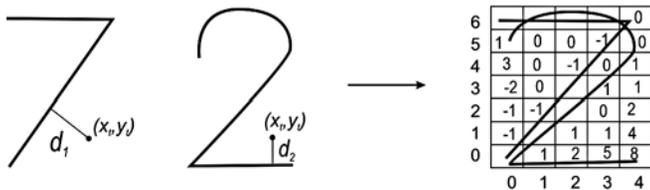


Рис. 8. Расстояния d_1 и d_2 для точки (x, y)

для разных эталонов будет различно, то в этом случае каждый слой образа кривой, для которой нет пары, может сравниваться с предыдущим слоем сравниваемого образа.

Другая возможность — это использование непосредственно графических образов без использования алгоритмов преобразования в кривые. В этом случае для каждой точки (или ячейки) поверхности изображения определяется кратчайшее расстояние до каждого графического образа. По значениям этих данных составляется таблица весов для каждой пары образов. На рис. 8 приведен пример для одной точки изображения (x_p, y_p) по отношению к графическим образам цифр "7" и "2".

Здесь для точки (x_p, y_p) наименьшие расстояния до каждого образа будут равны d_1 и d_2 . Соответственно вес w_{ij} для этой точки будет определяться как

$$w_{ij} = d_1^2 - d_2^2,$$

где d_1^2 и d_2^2 заменяют $(y_{etal1_j} - y_{test_{ij}})^2$, $(y_{etal2_i} - y_{test_{ij}})^2$ в выражении (4) и определяются соответственно выражениями

$$\begin{aligned} d_1^2 &= (x_1 - x_p)^2 + (y_1 - y_p)^2, \\ d_2^2 &= (x_2 - x_p)^2 + (y_2 - y_p)^2, \end{aligned} \quad (16)$$

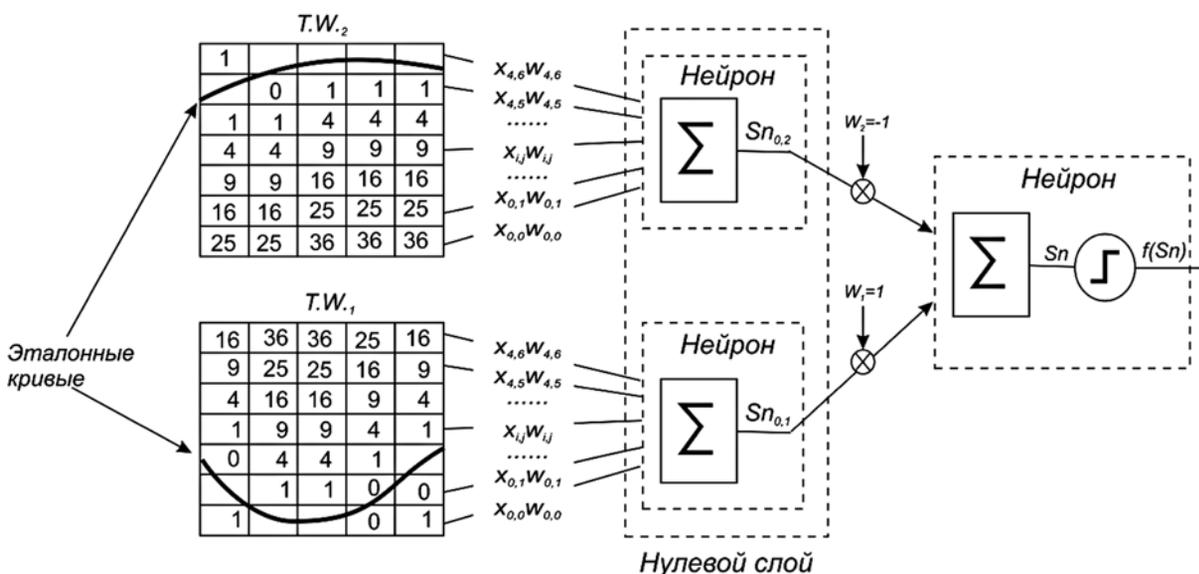


Рис. 9. Разделение нейрона первого слоя на два нейрона нулевого слоя

где (x_1, y_1) и (x_2, y_2) — ближайшие точки эталонных образов до точки (x_p, y_p) (рис. 8).

В целом логика работы сети будет выглядеть следующим образом. По аналогии со схемой на рис. 5 первый слой сети выполняет попарное сравнение всех эталонов, второй слой объединяет результаты первого слоя, соответствующие одному эталону в одном нейроне. Третий слой объединяет выходы эталонов одного образа в одном нейроне. Активный выход i -го нейрона третьего слоя означает соответствие распознаваемого объекта i -му образу. Такая логика нейронной сети соответствует алгоритму метода ближайшего соседа [5].

Нулевой слой нейронной сети

Для схемы на рис. 5 требуются вычисление таблиц весов в количестве, равном числу нейронов первого слоя, определяемому по выражениям (10), (12). Для того чтобы уменьшить число вычисляемых таблиц весов, разделим нейрон первого слоя на два нейрона, каждый из которых будет определяться таблицей весов одного эталона (рис. 9), где значение весов будет определяться в соответствии с (8), (15) формулами

$$w_{ij} = (y_{etal1_j} - y_{test_{ij}})^2; \quad (17)$$

$$w_{ij} = (x - x_p)^2 + (y - y_p)^2. \quad (18)$$

В итоге в нейронную схему, представленную на рис. 5, добавляется нулевой слой, где каждому нейрону соответствует один нейрон и соответственно одна таблица весов. При этом число вычисляемых таблиц весов уменьшается до числа эталонов (рис. 10).

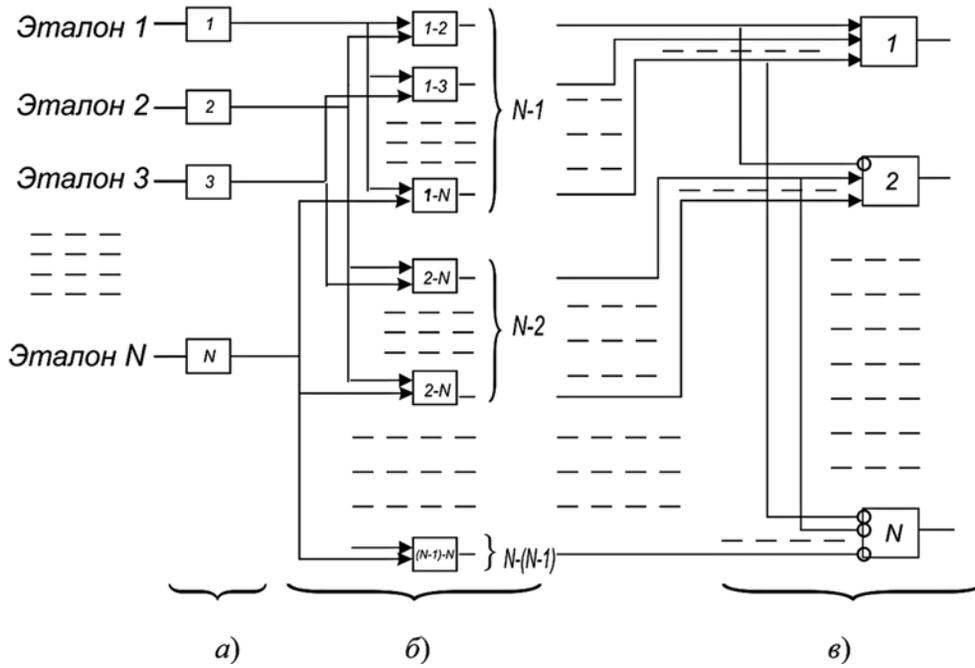


Рис. 10. Добавление нулевого слоя:
 а — нулевой слой; б — первый слой; в — второй слой

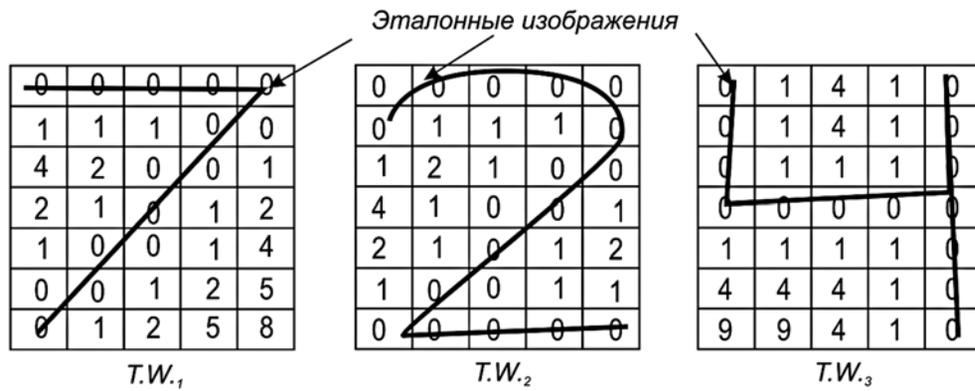


Рис. 11. Таблицы весов ($T.W.$) нулевого слоя для трех эталонов

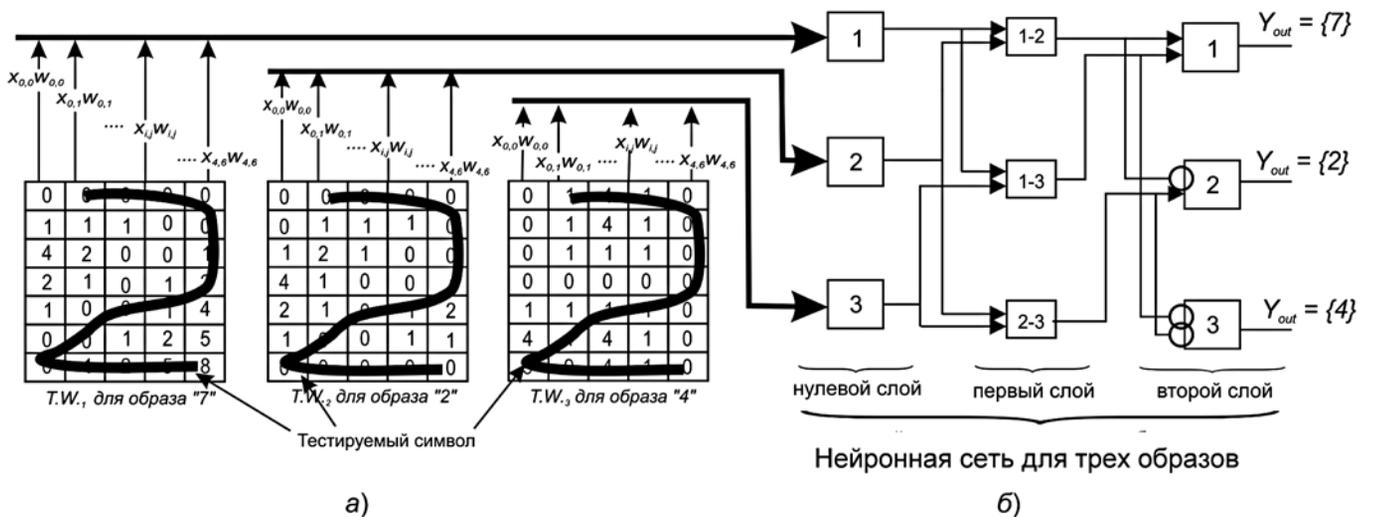


Рис. 12. Тестируемый символ (а) и нейронная сеть для трех эталонов (б)

Результат работы сети для схемы на рис. 12, б

Нулевой слой, k — порядковый номер нейрона и эталона			Первый слой, k и m — порядковые номера сравниваемых эталонов			Второй слой, k — номер нейрона и эталона			Y_{out}
k	$Sn_{0, k}$	$f(Sn_{0, k})$	k_m	$Sn_{1, k m}$	$f(Sn_{1, k m})$	k	$Sn_{2, k}$	$f(Sn_{2, k})$	
1	24	24	1-2	20	0	1	$0 + 1 = 1$	0	—
2	4	4	1-3	-8	1	2	$\bar{0} + 1 = 2$	1	{2}
3	32	32	2-3	-28	1	3	$\bar{1} + \bar{1} = 0$	0	—

Пример

Таблицы весов размерности 5:7 для выделенных эталонов образа "7", "2", "4" (рис. 11) определены на основе выражения (18), где в качестве координат x_i и y_i представляются номера ячейки по вертикали и горизонтали, а в качестве координат x и y определяются ближайшие ячейки, которые захватывают изображение символа. Каждому i -му выходу нейронной сети Y_{out} соответствует эталон с i -м порядком расположения на рис. 11 (например, на рис. 12, б первому выходу сети соответствует образ "7", второму выходу — "2", третьему — "4").

В табл. 1 представлен результат работы сети, приведенной на рис. 12, б, в результате тестирования тестируемого символа "2" (рис. 12, а), а также значение и выход каждого нейрона. По результатам табл. 1 можно видеть, что активизируется выход второго нейрона второго слоя, который соответствует образу символа {2}.

Отметим также, что в задаче распознавания эталонов в качестве функции, определяющей близость к эталону, может использоваться выражения, отличные от выражений (8), (15), (17), (18). В этом случае значения таблицы весов будут вычисляться согласно выражению, определяющему метрическую меру близости.

Алгоритм отбора эталонов

Набор эталонов может отбираться из существующей выборки либо произвольным образом, либо с помощью алгоритма отбора минимального набора эталонов, работа которого заключается в следующем. Изначально для каждого образа отбирается наилучший эталон. Для этого в базу данных поочередно назначаются элементы тестируемой выборки для данного образа. После каждого назначения выполняется полное распознавание для всех остальных элементов выборки, результаты которого фиксируются в виде кодовой строки, состоящей из нулей и единиц, где 1 соответствует идентифицированному элементу, 0 — не идентифицированному (см. табл. 1). Если число единиц в строке соответствует необходимой достоверности распознавания, то в качестве эталона для этого образа выбирается текущий элемент выборки, на этом алгоритм завершается. Если же после аналогичной процедуры для

всей выборки необходимый эталон не определяется, то выбирается кодовая строка с наибольшим числом единиц в строке, которая помещается в первую строку обучающей таблицы (табл. 2), а элемент, соответствующий этой строке, помещается в базу данных эталонов. Далее в полученной строке определяются номера столбцов, для которых значения равны нулю, что соответствует номерам элементов выборки, для которых распознавание было неудачно. В дальнейшем процедура повторяется аналогичным образом, но только теперь для этих не идентифицированных элементов. Каждый из этих элементов поочередно добавляется в базу эталонных данных, вычисляется кодовая строка, которая добавляется в обучающую таблицу (табл. 2). Итоговая полученная обучающая таблица становится объектом исследования на предмет определения наименьшего набора строк, поразрядное логическое сложение которых дает наибольшее число единиц в итоговой кодовой строке (табл. 2), значение которого должно быть не меньше минимально требуемого значения. В результате работы данного алгоритма получаем минимальный набор эталонов, применение которых обеспечивает необходимый уровень достоверности для используемой общей выборки эталонов. Нужно отметить, что данный алгоритм может применяться вне зависимости от того, какой набор признаков или какая характеристика меры близости используется в применяемом метрическом методе распознавания.

В табл. 2 приведен пример отбора набора эталонов для рукопечатного символа "А" (рис. 13) из выборки в 320 символов (рис. 14, а) (по 10 элементов на каждую букву алфавита). Распознавание

Таблица 2

Обучающая таблица для символа "А"

Добавляемые в базу символы	1	2	3	4	5	6	7	8	9	10	Сумма
<i>А — печ.</i>	1	0	1	0	1	1	1	0	1	1	7
<i>А-2</i>	1	1	1	0	1	0	1	1	1	0	7
<i>А-4</i>	1	1	0	1	1	1	0	1	1	0	7
<i>А-8</i>	1	0	1	1	1	0	0	1	1	1	7
<i>А — печ., А-4</i>	1	1	1	1	1	1	1	1	1	1	10

Заключение

Алгоритмы, реализующие метрические методы распознавания, преобразуются в нейронную сеть прямого распространения с определяемой и прозрачной архитектурой, а также с возможностью аналитически определять весовые значения связей полученной нейронной сети без использования классических алгоритмов обучения, что значительно ускоряет процесс создания нейронной сети. В качестве эталонов могут быть любые объекты. Набор и число эталонов может выбираться либо произвольным, либо эмпирическим способом, либо с помощью приведенного выше алгоритма отбора эталонов.

Способы представления эталонов могут быть различны (их выбирают сами разработчики сети), а также зависят от сложности решаемой задачи. В качестве эталонов могут быть как сами изображения объектов (см. рис. 7, 11), так и вектор параметров, полученный из самих изображений объектов образа и характеризующий определенный набор свойств распознаваемых образов. В этом случае эталон может быть представлен в виде изображения кривой (см. рис. 1), определяющей зависимость значений параметров вектора от их расположения в наборе параметров вектора. Если выделенные параметры распознаваемого объекта не равнозначны по своей значимости или по своей физической природе (как, например, температура, длина, масса и т. д.), то они могут быть также предварительно масштабированы.

Поскольку полученные нейронные сети (см. рис. 5, 10) являются сетями прямого распространения, то к данным сетям могут быть также при-

менены классические алгоритмы обучения [6–9]. В этом случае обычный процесс обучения фактически можно заменить на процесс до обучения. Даже с учетом процесса, возможного до обучения, вся процедура создания и обучения нейронной сети может значительно ускориться, поскольку весовые и пороговые значения уже определены и аналитически вычислены.

Список литературы

1. **Кульчин Ю. Н., Ким А. Ю., Ноткин Б. С., Люхтер А. Б.** Обработка сигналов распределенной волоконно-оптической сети для распознавания динамических образов с применением нейронных сетей // Информационные технологии. 2015. Т. 21. № 4. С. 312–319.
2. **Игнатъев Д. И., Храбров А. Н.** Нейросетевое моделирование нестационарных продольных аэродинамических характеристик самолета // Информационные технологии. 2014. № 3. С. 61–69.
3. **Дам В. Н.** Автоматическое распознавание цифровой модуляции радиосигналов с помощью многослойной нейронной сети по кумулянтным признакам // Информационные технологии. 2016. Т. 22. № 7. С. 555–560.
4. **Агеева У. О., Агеева В. Г., Барский А. Б.** Бионическое интеллектуальное протезирование конечностей и логические нейронные сети // Информационные технологии. 2016. Т. 22. № 5. С. 379–386.
5. **Биргер И. А.** Техническая диагностика. М.: Машиностроение, 1978, 240 с.
6. **Барский А. Б.** Нейронные сети: распознавание, управление, принятие решений. М.: Финансы и статистика, 2004. 176 с.
7. **Головко В. Л.** Нейронные сети: обучение, организация и применение / Под ред. А. И. Галушкина. Науч. сер. "Нейрокомпьютеры и их применение". Кн. 4. М.: ИПРЖР, 2001. 256 с.
8. **Розенблатт Ф.** Принципы нейродинамики: Перцептроны и теория механизмов мозга Principles of Neurodynamic: Perceptrons and the Theory of Brain Mechanisms. М.: Мир, 1965. 478 с.
9. **Уоссерман Ф.** Нейрокомпьютерная техника. Теория и Практика. [Рус. пер.]. М.: Мир, 1992. 97 с.

P. Sh. Geidarov, Phd, Associate Professor, e-mail: plbaku2010@gmail.com,
Institute for System Control NAS of Azerbaijan, Baku

The Feedforward Neural Network with Calculated Parameters

The paper describes the possibility of creating the feedforward neural networks with a clear defined architecture and with calculated parameters, realizing the metric methods of recognition. For these networks number of neurons, connections, as well as weights and thresholds values are calculated analytically, based on the initial conditions of tasks: number of recognizable classes, number of samples, metric expressions used. The work also shows ways of calculating the value of tables of weights without the use of classical learning algorithms. We also consider the examples of the calculation tables of weights of the neural network connections. Also, the work suggests ways to reduce the number of calculations. For this purpose, changes are made in the neural network architecture. The article also discusses the possibility of minimizing the number of neurons and connections to networks received. In the paper also describes the algorithm that implements the selection of a minimum set of samples from learning set. The received conclusions are checked on examples with handwritten symbols and figures.

Keywords: models of neural networks, neural network training, linear neural network, nearest neighbor method, the method of sample, metric methods of recognition, activation function

References

1. **Kul'chin Ju. N., Kim A. Ju., Notkin B. S., Ljuhter A. B.** Obrabotka signalov raspredelennoj volokonno-opticheskoj seti dlja raspoznavanija dinamičeskikh obrazov s primeneniem nejronnyh setej (Signal Processing System of DFMN Segmental Type for Recognition of Dynamic Images Using Neural Networks), *Informacionnye tehnologii*, 2015, vol. 21, no. 4, pp. 312–319 (in Russian).
2. **Ignat'ev D. I., Hrabrov A. N.** Nejrosetevoe modelirovanie nestacionarnyh prodol'nyh ajerodinamičeskikh harakteristik samoleta (Modeling of Unsteady Longitudinal Aerodynamic Characteristics of Aircraft Using Neural Networks), *Informacionnye tehnologii*, 2014, no. 3, pp. 61–69 (in Russian).
3. **Dam V. N.** Avtomatičeskoe raspoznavanie cifrovoj moduljaccii radiosignalov s pomoshh'ju mnogoslojnoj nejronnoj seti po kumuljantnym priznakam (Automatic Digital Modulation Recognition of Radio Signals Using Multilayer Neural Network with Cumulant Features), *Informacionnye tehnologii*, 2016, vol. 22, no. 7, pp. 555–560 (in Russian).
4. **Ageeva U. O., Ageeva V. G., Barskij A. B.** Bioničeskoe intellektual'noe protezirovanie konečnej i logičeskie nejronnye seti (Bionic Intelligence Limbs Prosthetic and Logical Neural Networks), *Informacionnye tehnologii*, 2016, vol. 22, no. 5, pp. 379–386 (in Russian).
5. **Birger I. A.** *Tehničeskaja diagnostika* (Technical diagnostics), Moscow, Mashinostroenie, 1978. 240 p. (in Russian).
6. **Barskij A. B.** *Nejronnye seti: raspoznavanie, upravlenie, prinjatie reshenij* (Neural networks: recognition, management, decision-making), Moscow, Finansy i statistika, 2004. 176 p. (in Russian).
7. **Golovko V. L.** *Nejronnye seti: obučenie, organizacija i primenenie* (Neural networks: education, organization and application), Moscow, 2001. 256 p. (in Russian).
8. **Rozenblatt F.** *Principy nejrodinamiki: Perceptrony i teorija mehanizmov mozga* (Principles of Neurodynamic: Perceptrons and the Theory of Brain Mechanisms), Moscow, Mir, 1965. 478 p. (in Russian).
9. **Wasserman F. D.** *Nejrokompjuternaja tehnika. Teorija i Praktika* (Neural Computing: Theory and Practice), Moscow, Mir, 1992. 97 p. (in Russian).

Адрес редакции:

107076, Москва, Стромьинский пер., 4

Телефон редакции журнала (499) 269-5510

E-mail: it@novtex.ru

Технический редактор *Е. В. Конова*.

Корректор *Н. В. Яшина*.

Сдано в набор 11.05.2017. Подписано в печать 22.06.2017. Формат 60×88 1/8. Бумага офсетная.

Усл. печ. л. 8,86. Заказ IT717. Цена договорная.

Журнал зарегистрирован в Министерстве Российской Федерации по делам печати, телерадиовещания и средств массовых коммуникаций.

Свидетельство о регистрации ПИ № 77-15565 от 02 июня 2003 г.

Оригинал-макет ООО "Авансед солюшнз". Отпечатано в ООО "Авансед солюшнз".

119071, г. Москва, Ленинский пр-т, д. 19, стр. 1.
