

СЕКЦИЯ 9

**ФИНИШНАЯ ПРЯМАЯ:  
СЕКЦИЯ АСПИРАНТОВ**



УДК 681.3

**УПРАВЛЯЕМЫЙ ПОСТБИНАРНЫЙ КЛЕТОЧНЫЙ АВТОМАТ****Аноприенко А.Я., Коноплева А.П.**

Донецкий национальный технический университет

Кафедра компьютерной инженерии

E-mail: [anna.konoplyova@gmail.com](mailto:anna.konoplyova@gmail.com)**Аннотация**

*Аноприенко А.Я., Коноплева А.П. Управляемый постбинарный клеточный автомат. Доклад посвящен управляемому постбинарному клеточному автомату, который отличается от своих предшественников – клеточного автомата (КА) Конвея и неуправляемого постбинарного клеточного автомата (ПКА) наличием пользовательских элементов управления процессом эволюции. Подробно описан интерфейс управляемого ПКА. Выполнено сравнение алгоритмов работы управляемого ПКА с КА Конвея и неуправляемым ПКА. Исследован процесс эволюции управляемого ПКА в зависимости от параметров, задаваемых пользователем.*

**Общая постановка проблемы**

Клеточные автоматы (КА) широко применяются для моделирования и изучения динамических процессов в разных областях научного знания, включая физику, химию, биологию, социологию и др [1-4].

Постбинарные клеточные автоматы (ПКА) – это вид КА, в котором исходные комбинации задаются с помощью гиперкодов, а при задании состояний клеток и локальных зависимостей используются элементы гиперлогики. Их главное назначение – расширить возможности моделирования посредством КА [5-10]. В контексте данной статьи такие ПКА будем называть неуправляемыми ПКА (НПКА).

Задача, которая решается в данной работе – разработка и исследование возможностей управляемого ПКА (УПКА), т.е. такого автомата, законы-рецепты которого пользователь может менять в процессе моделирования, что может значительно повлиять на процесс исследования изучаемого посредством КА явления.

**Интерфейс управляемого ПКА**

Для решения поставленной задачи немаловажным вопросом является разработка грамотного функционального интерфейса, который был бы пользователю не только удобен и понятен, но и содержал в себе возможности визуального отображения задаваемых пользователем правил. Данный вариант ПКА был разработан в среде Flash на языке ActionScript. Это дает возможность работать с данным продуктом непосредственно в он-лайн режиме через Интернет.

Интерфейс управляемого ПКА содержит панель управления процессом работы автомата и непосредственно поле, на котором осуществляется моделирование (рис. 1).

По сравнению с предыдущими версиями ПКА [5-10] в интерфейс добавлены новые компоненты. Пользователь имеет возможность задавать следующие параметры ПКА:

- тип соседства (Мура или фон Неймана);

- условия рождения трех видов клеток регулируется 2-мя параметрами: нижними границами, по которым определяются количество «суперклеток» и количество «больных клеток»;

- условия выживания и заболевания клеток (пользователь должен определить нижнюю границу, по которой будет вычисляться количество «больных клеток» и количество «суперклеток»);

- распараллеливание значений (для случая заболевания или выздоровления клеток). Распараллеливание осуществляется на всех живых соседей данной клетки.

Кроме того, для каждого описанного выше параметра существует поле, в котором пользователь может просмотреть различные варианты задаваемых им правил. Так, например, в программе предусмотрено отображение 2-х типов соседства, т.е. набор соседних клеток, которые могут влиять на данную: соседство Мура и соседство фон Неймана (рис. 2).

Для просмотра возможных вариантов, при которых, при заданных параметрах “number of super” и “number of sick” (количество «суперклеток» и количество «больных клеток»), может родиться «нормальная клетка» или «суперклетка» или «больная клетка», пользователь должен выбрать соответствующую “radio button” (рис.3).

Все описанные выше параметры могут менять ход эволюции КА в процессе его работы. Пользователю для этого достаточно нажать «stop» и поменять важный для него параметр.

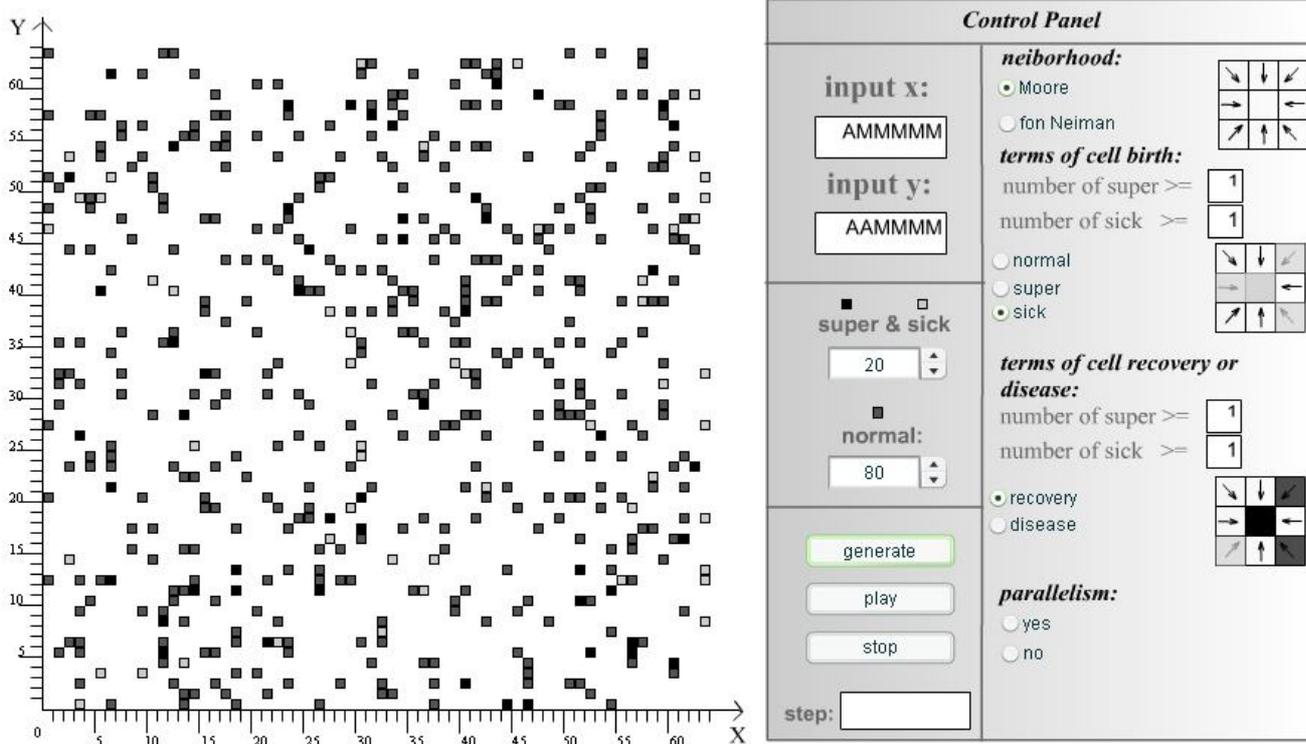


Рис.1. Интерфейс управляемого постбинарного клеточного автомата

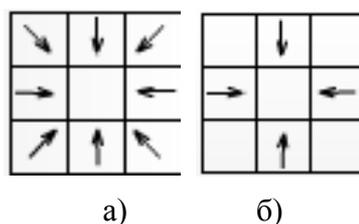


Рис. 2. Предусмотренные программой типы соседства: а) соседство Мура, б) соседство фон Неймана

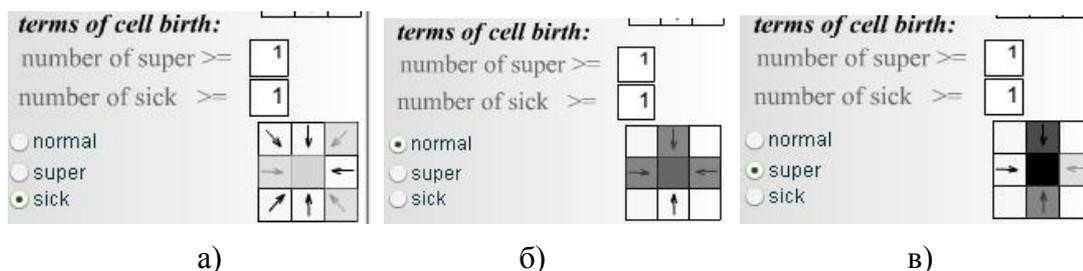


Рис.3. Отображение возможных вариантов расположения и качества соседних клеток, при котором может родиться а) «больная клетка», б) «нормальная клетка», в) «суперклетка»

### Сравнение алгоритма работы управляемого ПКА, традиционного КА Конвея и неуправляемого ПКА

При разработке ПКА был взят за основу алгоритм клеточного автомата Конвея (игра «Жизнь»)[1-4]. Однако, на очередном этапе работы в традиционный алгоритм были внесены существенные изменения, благодаря новым возможностям для моделирования, которые дают гиперлогику и гиперкоды [5-10], что привело к появлению нового типа КА – ПКА.

Правила работы управляемого ПКА зависят от параметров, задаваемых пользователем. Таким образом, алгоритм работы такого автомата с одной стороны наследует некоторые принципы двух описанных выше КА («Жизнь» Конвея и ПКА), а с другой – значительно отличается как от традиционного КА Конвея, так и от ПКА.

Набор правил такого ПКА вербально можно описать следующим образом:

- *Рождение клеток:*

- как и в случае традиционного автомата Конвея, клетка может появиться в случае наличия трех соседей;
- как и в ПКА могут родиться 3 вида клеток: «нормальная клетка», «суперклетка», «больная клетка»;
- «суперклетка» рождается, если количество «супер-соседей» больше либо равно заданного пользователем («number of super»), иначе если количество «больных соседей» больше либо равно заданного («number of sick»), рождается «больная клетка», и только, если оба условия не удовлетворены, рождается «нормальная клетка».

- *Выздоровление клеток:*

- данный вариант развития алгоритма не предусматривался Конвеем, но присутствует в ПКА;
- в случае с управляемым ПКА для пользователя предусмотрена возможность выбора: распараллеливать выздоровление или нет: может выздоравливать либо просто единственная  $i$ -ая клетка (если пользователь выбрал опцию «по» в разделе «parallel» – без распараллеливания), либо происходит выздоровление всех клеток-соседей для  $i$ -ой (если

пользователь выбрал опцию «yes» в разделе «parallel» – с распараллеливанием).

- выздоровление происходит в случае, когда количество «супер-соседей» больше либо равно задаваемых пользователем «number of super».

- *Заболевание клеток:*

- данный вариант развития алгоритма (как и выздоровление) не предусматривался Конвеем, но присутствует в ПКА;
- в случае с управляемым ПКА для пользователя также предусмотрена возможность выбора: распараллеливать заболевание или нет. Может заболеть либо просто единственная *i*-ая клетка (если пользователь выбрал опцию «no» в разделе «parallel» – без распараллеливания), либо происходит заболевание всех клеток-соседей для *i*-ой (если пользователь выбрал опцию «yes» в разделе «parallel» – с распараллеливанием)
- заболевание происходит в случае, когда количество «больных соседей» больше либо равно задаваемых пользователем «number of sick».

- *Гибель клеток:*

Правило аналогично автомату Конвея и неуправляемому ПКА: клетка гибнет, если количество соседей меньше двух или больше трех.

### **Исследование управляемого ПКА**

Используя параметры, которые можно менять в процессе эволюции автомата, пользователь может управлять работой автомата, что дает ему возможность наблюдать некоторые новые свойства ПКА:

- тип соседства фон Неймана является в несколько раз менее эффективным для данного автомата, чем соседство Мура, поскольку генерация клеток вымирает через несколько шагов, либо застывает в статичном состоянии;

- повышение приоритета рождаемости «слабых клеток» над «супер клетками» также приводит либо к гибели всей генерации клеток, либо к статичным конфигурациям за несколько шагов;

- без распараллеливания выздоровления и заболевания клеток за несколько шагов генерация клеток сжимается либо в небольшую статичную конфигурацию, либо погибает;

- при повышении уровня заболеваемости клеток (повышение приоритета «больных клеток» над «суперклетками») в процессе эволюции начинает возрастать генерация больных клеток, что приводит к гибели всей генерации или установлению ее в статичное состояние;

- повышенный уровень приоритета рождаемости «супер клеток» над «больными клетками» дает более интересную и длительную эволюцию генерации;

Это лишь некоторые выводы, которые пользователь может сделать при исследовании данного КА, без длительных вычислений и доказательств, просто меняя те или иные параметры в программе. Это наглядно и понятно прослеживается на поле эволюции.

Таким образом, очевидным преимуществом управляемого ПКА является существенное расширение возможностей пользователя, и повышение уровня наглядности в процессе исследований.

### **Выводы**

В данном докладе предложен новый тип ПКА – управляемый постбинарный клеточный автомат. Решается задача разработки и исследования возможностей управляемого ПКА, которые предоставляются пользователю в процессе его использования. Усложняется алгоритм работы КА в зависимости от вводимых параметров. Предыдущие версии традиционного автомата Конвея и разработанные авторами версии ПКА не давали возможности управления процессом эволюции.

Существенным преимуществом данного ПКА, является возможность управления процессом во время эволюции генерации. Это существенно улучшает эффективность исследования и повышает наглядность программы при моделировании реальных процессов, например, таких как: развитие популяции, распространение инфекций, пожаров и т.п.

Таким образом, можно сделать вывод о полезности внедрения элементов управления процессом эволюции в клеточные автоматы и постбинарные клеточные автоматы.

### Литература

1. Аладьев В.З., Бойко В.К., Ровба Е.А Классические однородные структуры. Теория и приложения: монография / – Гродно : ГрГУ, 2008. – 486 с. – ISBN 978-985-515-020-7. – ISBN 978-9985-9508-4-5.
2. Тофоли Т., Марголус Н. Машины клеточных автоматов // Издательство “Мир” Москва, 1991. – 280 с.
3. Wolfram S. A New Kind of Science // [Электронный ресурс] / Wolfram S. Режим доступа – <http://www.wolframscience.com/nksonline/toc.html>
4. Гормакова И.В. Вычислительные свойства сетей клеточных автоматов // Вестник национального технического университета «ХПИ». Сборник научных трудов. Тематический выпуск «Автоматика и приборостроение». Выпуск 56’2008 [Электронный ресурс] / Гормакова И.В. Режим доступа – [http://library.kpi.kharkov.ua/Vestnik/2008\\_56.pdf](http://library.kpi.kharkov.ua/Vestnik/2008_56.pdf)
5. Аноприенко А.Я., Коноплева А.П. Опыт применения гиперкодов в моделировании клеточных автоматов // Научные труды Донецкого национального технического университета. Серия "Проблемы моделирования и автоматизации проектирования динамических систем" (МАП-2007). Выпуск 6 (127): Донецк: ДонНТУ, 2007. С. 220-227.
6. Аноприенко А.Я., Коноплева А.П. Развитие идеи применения гиперкодов в моделировании клеточных автоматов // Научные труды Донецкого национального технического университета. Серия: Информатика, кибернетика и вычислительная техника (ИКВТ-2008) выпуск 93: - Донецк: ДонНТУ, 2008. С. 289-316.
7. Аноприенко А.Я., Коноплева А.П., Василенко А.Ю. Оценка производительности при моделировании постбинарных клеточных автоматов и способы ее повышения // Научные труды Донецкого национального технического университета. Серия: Информатика, кибернетика и вычислительная техника (ИКВТ-2009) выпуск 147: - Донецк: ДонНТУ, 2009. С. 96-104.
8. Аноприенко А. Я., Коноплева А. П. Клеточные автоматы в историческом контексте и их классификация // Збірка матеріалів п'ятої міжнародної науково-технічної конференції студентів, аспірантів та молодих науковців 2009 р.. Серія «Інформатика та комп'ютерні технології» (ІКТ-2009). – Донецьк: ДонНТУ. – 2009. – С.635-640.
9. Аноприенко А. Я., Коноплева А. П. Моделирование постбинарных клеточных автоматов // Збірник наукових праць. Спеціальний випуск. Матеріали міжнародної наукової конференції «Моделювання 2010» (12-14 травня 2010 р.). Том 2., - Київ: Інститут проблем моделювання в енергетиці ім. Г.Є. Пухова. НАН України. 2010. С. 162-170
10. Коноплева А.П. Способы формального описания классических и постбинарных клеточных автоматов // Наукові праці Донецького національного технічного університету. Випуск 9 (149). Серія «Проблеми моделювання та автоматизації проектування». – Донецьк: ДонНТУ – 2011. С. 184-194.

УДК 004.031.42:37

## **ВЕБ-ОРИЕНТОВАНА КОМП'ЮТЕРНА СИСТЕМА ПІДТРИМКИ НАВЧАЛЬНОГО ПРОЦЕСУ ПІДВИЩЕННЯ КВАЛІФІКАЦІЇ**

**Валуйський В.М., Гончаренко М.В., Павловський А.А., Безпалый Є.О.**

Національний Технічний Університет України «Київський Політехнічний Інститут»,  
кафедра спеціальних комп'ютерних систем  
E-mail: [pavlovskiy.anatoliy@gmail.com](mailto:pavlovskiy.anatoliy@gmail.com)

### **Анотація**

*Валуйський В.М., Гончаренко М.В., Павловський А.А., Безпалый Є.О. Веб-орієнтована комп'ютерна система підтримки навчального процесу підвищення кваліфікації. Представлено розробку веб-орієнтованої комп'ютерної системи підтримки навчального процесу підвищення кваліфікації співробітників вищих навчальних закладів, показані переваги такої системи і визначено її місце у веб-орієнтованому навчальному просторі. Приведені характеристики модулів системи, принципи їх функціонування та взаємодії із іншими системами, що використовуються в навчальному процесі. Дані рекомендації до подальшого удосконалення системи.*

### **Вступ**

Підрозділи підвищення кваліфікації сучасних вищих навчальних закладів (ВНЗ), як правило, використовують сучасні інформаційні веб-орієнтовані технології для підтримки навчального процесу (платформи підтримки навчання, системи управління змістом (Content Management Systems - CMS), веб-портали тощо. Різноманітність ресурсів і сервісів, що використовуються в такому комплексі, потребують складної системи підтримки, управління тощо. Важливість запропонованої системи зростає при впровадженні веб-орієнтованих навчальних просторів [1] (рис.1), що є невід'ємною складовою сучасної системи освіти.

**Мета роботи** – рекомендувати розробку та дати рекомендації щодо створення комп'ютерної системи підтримки навчального процесу підвищення кваліфікації у ВНЗ, яка б забезпечувала реалізацію модулів управління типової схеми навчального веб-простору, і взаємодіючи із сервісами підтримки навчання, реалізувала сучасний комплекс підтримки навчального процесу підвищення кваліфікації.

### **Аналіз існуючих рішень**

При аналізі існуючих рішень розглянуті системи планування й аналізу виробничої діяльності підприємства (*ERP - Enterprise Resource Planning* (Планування Ресурсів Підприємства) та *CRM - Customer Relationship Management* (Система управління взаємовідносин з клієнтами)).

Актуальність таких систем обумовлюється тим, що їх програмне забезпечення дозволяє управляти, оптимізувати, одержувати аналітичну інформацію щодо основної діяльності підприємства в цілому.

Прикладом впровадження *ERP* систем у ВНЗ є інструмент для управління навчальним процесом і всією фінансово-господарською діяльністю ВНЗ – “Галактика *ERP*” [2]. Рішення призначене освітнім установам, які володіють аудиторним фондом, контингентом професорсько-викладацького та студентського складу, готують фахівців в області як вищої, так і післявузівської (аспірантура, перепідготовка) освіти. До основних модулів даної системи можна віднести: модуль автоматизованого навчального процесу; модуль прийомної

комісії; контингент учнів; штатний розклад і кадри; держконтракти і договори; планування і облік фінансів; спеціалізовані рішення і інтеграція; безпека та адміністрування.

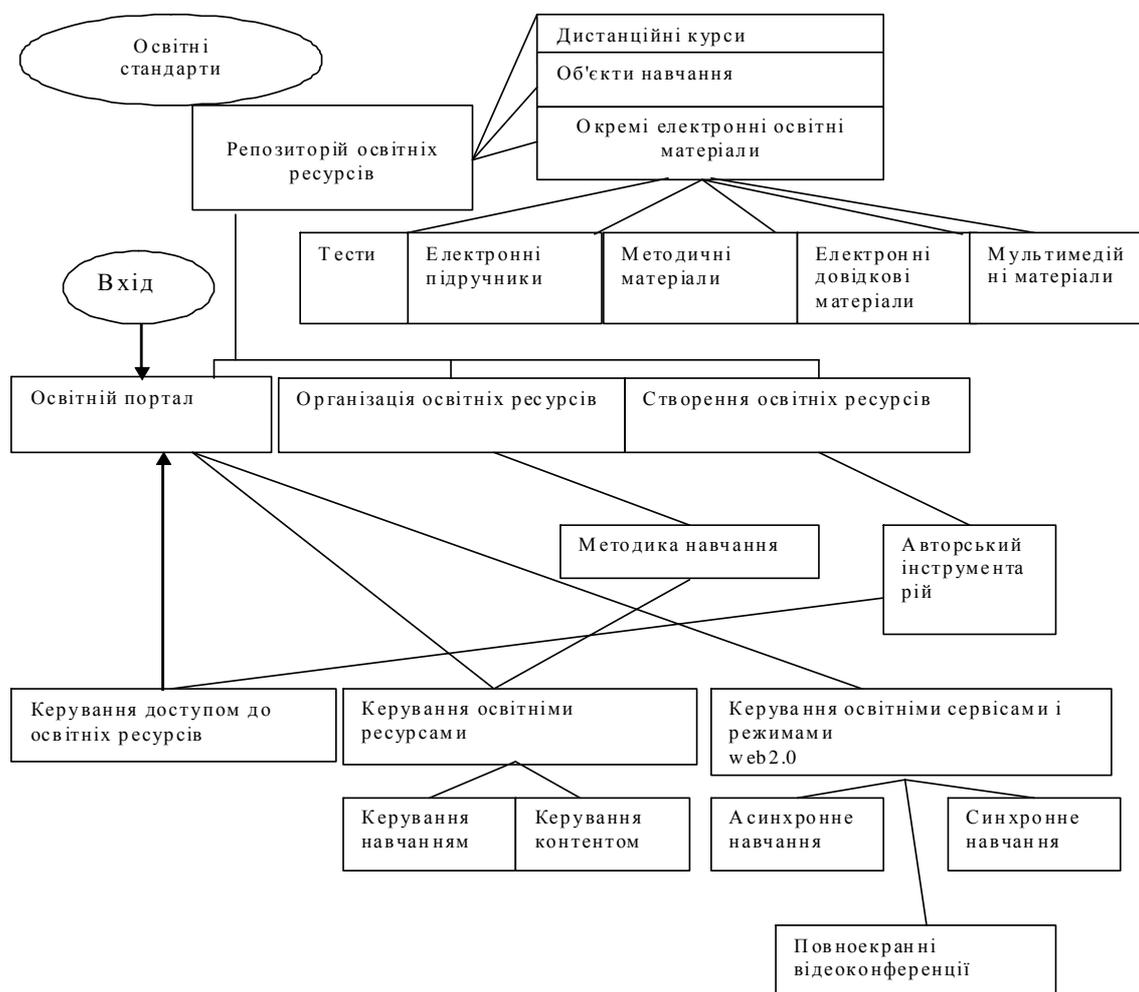


Рис 1. Типова схема навчального веб-простору.

Першим ВНЗ на території СНГ, який використовує *CRM* систему, став Російський державний відкритий технічний університет шляхів сполучення [3]. У навчальний процес він інтегрував *CRM* систему фірми *ASoft*. Основними перевагами даної системи є: оперативний доступ до єдиної інформаційної бази; планування власного робочого часу і діяльності співробітників; управління процесами (продажі, підтримки, закупівлі та інші); робота з документами; отримання якісних і кількісних звітів.

Розглянуті системи є ресурсомісткими, потребують складного процесу інтеграції, та значних фінансових капіталовкладень. Кожна з них містить надлишок функцій при використанні в процесі підвищення кваліфікації. Навчальний процес підвищення кваліфікації співробітників ВНЗ, окремий унікальний напрямок загального навчального процесу, що потребує індивідуального підходу при створенні спеціалізованого програмного забезпечення. Також важливим є інтеграція системи із вже встановленими платформами дистанційної освіти, системами управління змістом, системами єдиного доступу тощо. Детальний аналіз прототипів показав, що на даний момент, на інформаційному ринку немає спеціалізованих програмних систем для підтримки навчального процесу підвищення кваліфікації, а існуючі *ERP* та *CRM* системи мають велику вартість і надлишок функцій, та не в змозі задовольнити всі потреби даної галузі.

### Призначення та цілі

Основною ціллю комп'ютерної системи підтримки процесу підвищення кваліфікації є створення програмного продукту, який поєднає компоненти веб-освітнього простору, та реалізує функції: реєстрації слухачів у системі та внесення їх до системи каталогів[4]; підтримки єдиного доступу Single Sign On[5]; формування груп та черг на навчання; управління курсами підвищення кваліфікації; вибору і закріплення аудиторій за курсом, календаря; отримання якісних звітів; моніторингу контролю успішності, зберігання результатів; інформування викладача та слухачів.

Система повинна мати систему ролей, тісно взаємодіяти із іншими системами, що використовуються в процесі навчання, і мати модульну систему реалізації.

Система призначена для впровадження, як компонент веб-орієнтованого навчального простору, у підрозділах ВНЗ, що забезпечують перепідготовку та підвищення кваліфікації.

#### Основні компоненти системи

Система комп'ютерної підтримки процесу підвищення кваліфікації має ролі адміністратора (координатора) навчання і користувача (рис. 2).

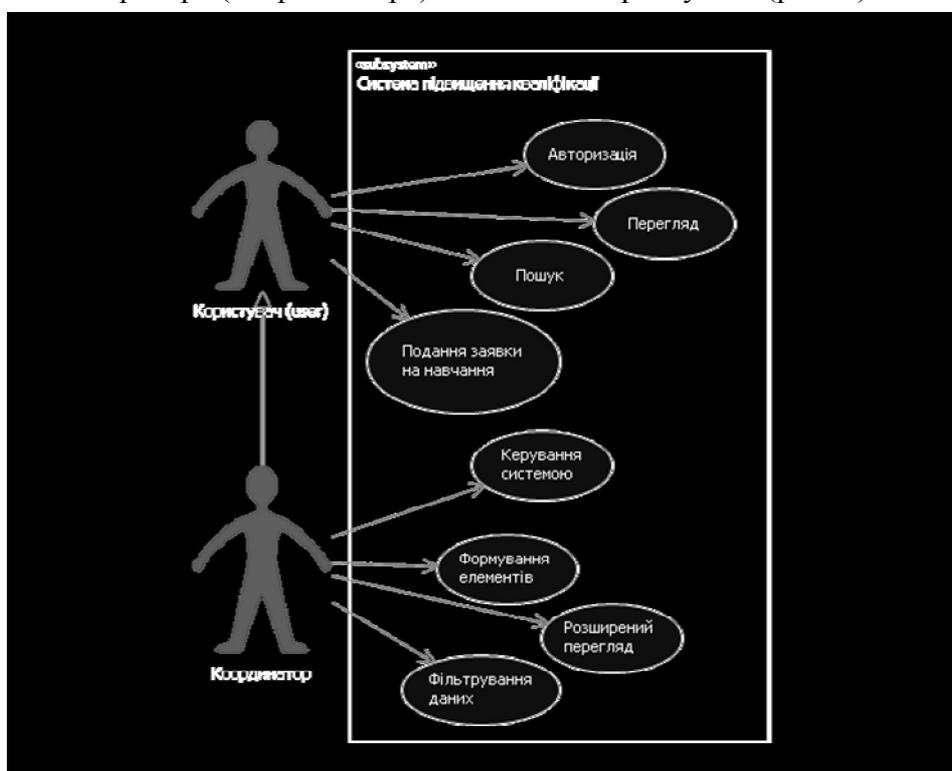


Рис 2. Діаграма прецедентів системи підтримки підвищення кваліфікації.

Роль користувача використовується після реєстрації студента або викладача, несе інформаційний зміст та надає можливість подавати заявки на навчання за іншими курсами, які пропонує ВНЗ. Викладач отримує доступ до персонального кабінету, де має можливість переглянути свій розклад, завантажити чи експортувати його до іншої системи планування подій.

Система складається із модулів: «Черга», «Слухачі», «Групи», «Курси», «Картки», «Звіти», «Викладачі», «Організації», «Календар». Всі модулі взаємодіють між собою.

Одним із ключових модулів в процесі підвищення кваліфікації є модуль «Черга». Його головне призначення – попередня реєстрація слухачів на навчання за одним з напрямків курсу (рис 3). Нові або існуючі слухачі, після подання заявки, потрапляють до черги на навчання. Головною можливістю модуля є те, що в ньому координатор має можливість розподіляти слухачів по групах. Коли група сформована – координатор може відправити її у процес навчання на певний курс. Основними функціями модуля є: реєстрація

нового слухача в системі; створення різних видів груп (процесної або наказної); занесення слухачів до групи та переміщення між ними; відправлення групи до процесу навчання.

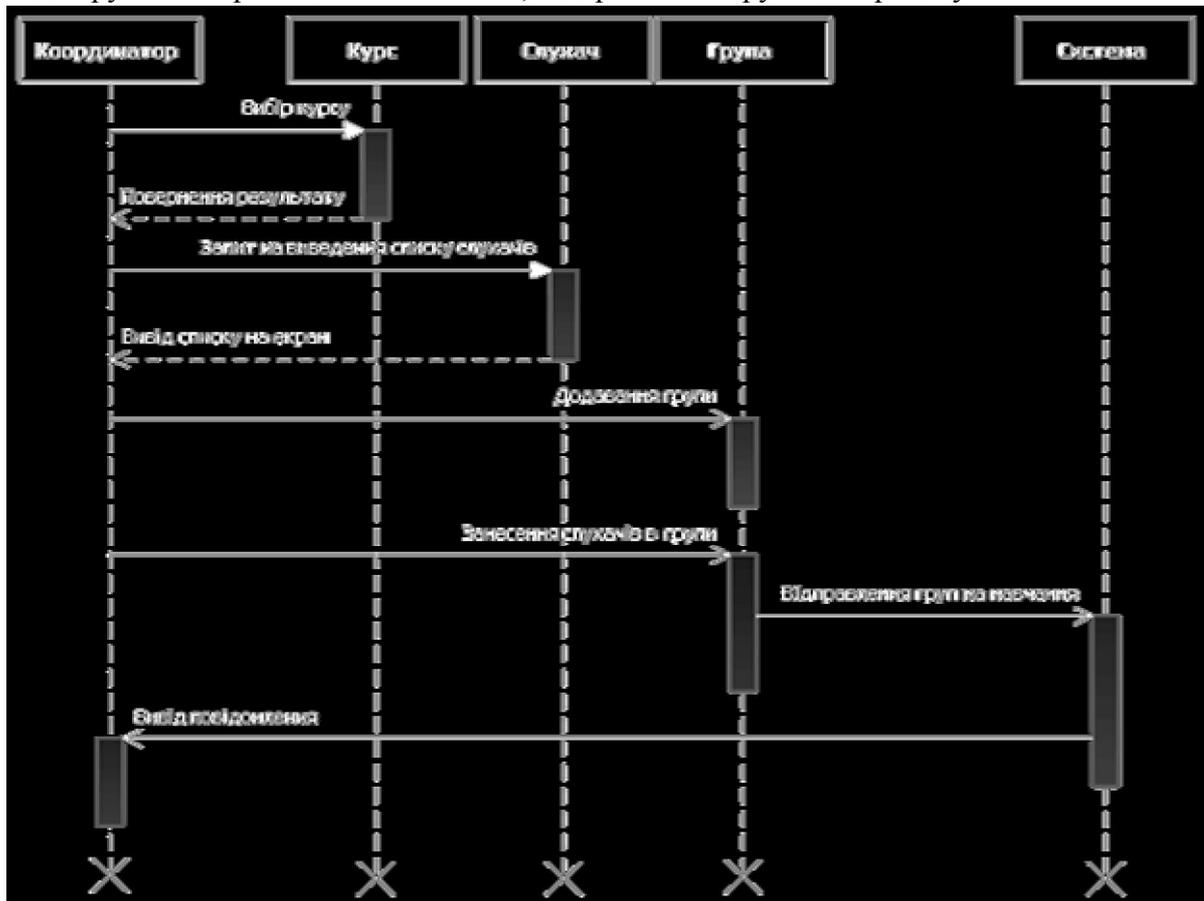


Рис 3. Діаграма послідовності відправлення груп на навчання.

Взаємодіючи із модулями «Групи» і «Курси», в залежності від типу групи і курсу, по якому ведеться навчання, дані про слухача заносяться до внутрішньої бази чи дерева каталогів LDAP. Після відправлення групи на навчання модуль автоматично розсилає повідомлення на електронні адреси слухачів, про формування відповідної групи з датами початку та закінчення навчання.

Головною метою модулю «Слухачі» є керування списком слухачів, які проходять підвищення кваліфікації та зареєстровані в системі. Модуль дозволяє редагувати інформацію про слухача, видаляти його, виводити списки слухачів, що навчаються за різними параметрами пошуку (група, П.І.Б., слухачі із заліком, але без свідоцтва тощо.) В режимі користувача модуль дозволяє переглянути / змінити персональні дані та надає слухачеві можливість перегляду свого розкладу.

Модуль «Групи» призначений для управління списком груп, які зареєстровані у навчальному процесі. В ньому виставляється тип групи, проводиться генерування логінів і паролів для слухачів групи із занесенням їх в систему каталогів чи внутрішню базу даних. Взаємодіючи із модулем «Календар», формується розклад та закріплюється викладач. Також функціями даного модуля є: задання фільтру пошуку з умовами (назва, курс, тип групи тощо) та вивід результатів; реєстрація, редагування або видалення групи з системи; друкування або завантажування списку слухачів групи.

В режимі користувача надається можливість дізнатись назву групи, аудиторію, продивитись список одногрупників тощо.

Головною метою модуля «Курси» є управління списком курсів, по яким здійснюється навчання за напрямком підвищення кваліфікації. Модуль дозволяє створювати нові курси,

вносити навчальну програму, короткий опис, необхідний рівень підготовки. Також за курсом закріплюються аудиторії, в яких є технічна спроможність провести заняття по вибраному напрямку. Якщо при навчанні використовується СДО[6], то курс закріплюється за відповідним курсом чи ресурсом із банку навчальних веб-ресурсів (рис 4.).

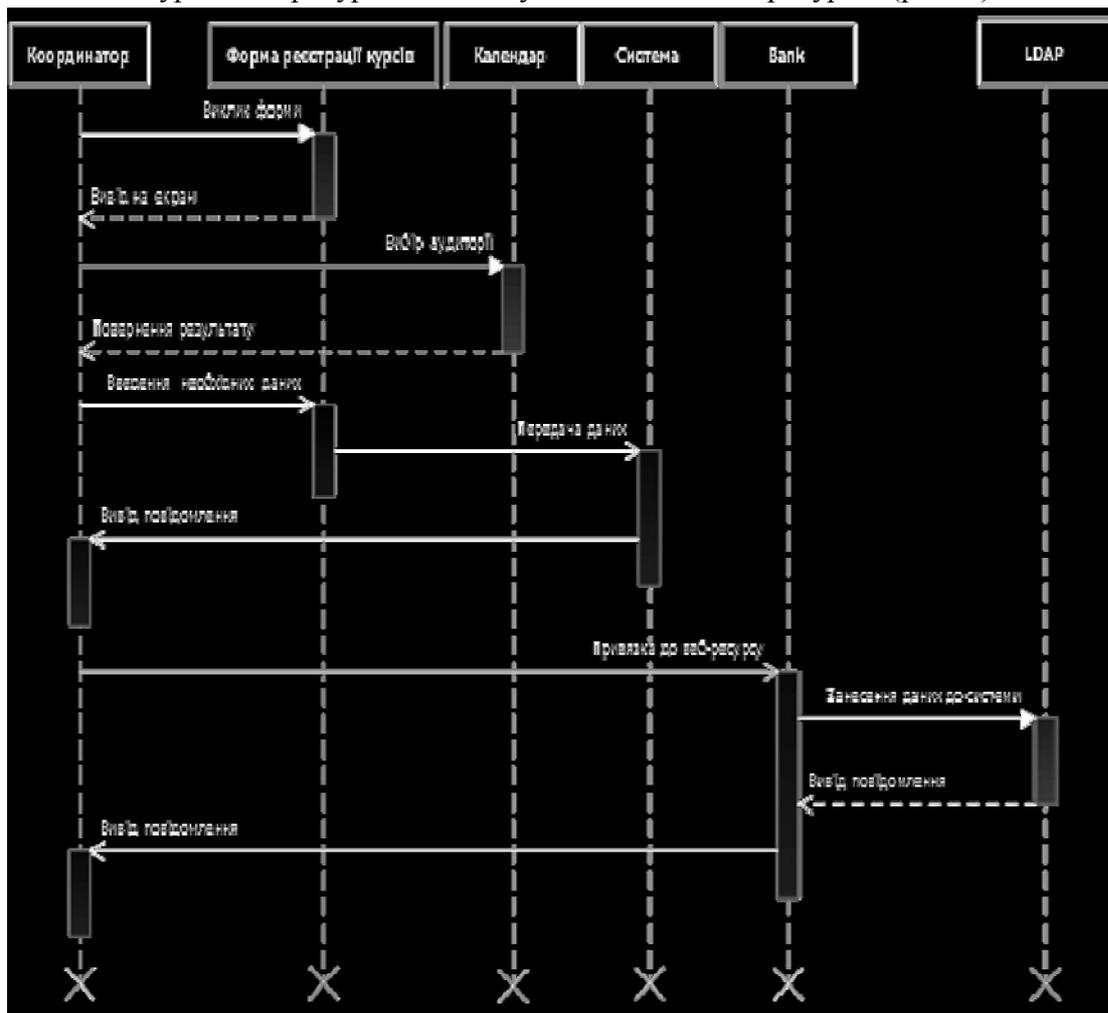


Рис 4. Діаграма послідовності реєстрації курсу у Системі

При зв'язуванні із банком дані також заносяться до дерева каталогів. Серед допоміжних функцій модуля є: перегляд програми курсу, завантаження або її друк, перегляд існуючих курсів, тощо.

Головною метою модулю «Викладачі» є управління списком викладачів, які викладають курси по підвищенню кваліфікації, закріплення їх за відповідними курсами та збереження відповідної інформації в системі каталогів.

Основними функціями модуля є: вивід списку викладачів; додавання викладачів в систему та інформації про них; редагування або видалення із системи, а взаємодіючи із модулем «Календар»: перегляд сформованого розкладу викладача; друкування розкладу, інформації про викладача тощо.

В користувацькому режимі викладача, модуль забезпечує доступ до персонального кабінету викладача для перегляду свого розкладу. В подальшому планується розширення даного модуля реалізації персонального кабінету на основі персонального веб-сайту викладача.

Модуль «Календар» дозволяє ефективно використовувати аудиторський фонд та викладацький склад. Модуль тісно зв'язаний із іншими модулями системи. Дозволяє переглядати завантаженість аудиторій, розклад викладачів, надає можливість підбирати дати проведення, закінчення курсу в залежності від навантаженості аудиторій, що технічно

підготовленні для нього. Серед допоміжних функцій - можливість експорту вибраного календаря для зовнішніх програм планування часу.

Модуль «Організації» дозволяє координаторам підготовки додавати до бази даних нові організації, співробітники яких проходять підготовку в даному навчальному закладі. Модуль дозволяє вводити та редагувати всю службову інформацію по вибраній організації. Подальший розвиток модуля передбачає його зв'язок із системами бухгалтерського обліку для надання чи прийому і передачі рахунків за навчання, передачі звітів, зв'язку із системами електронного обороту тощо.

#### **Реалізація системи**

Система реалізована із використанням мови програмування PHP, використовує внутрішню базу даних MySQL, та активне дерево каталогів LDAP і веб-сервер Apache. Використання цих технологій дало можливість тісно інтегрувати систему підтримки процесу підвищення кваліфікації з відкритими програмними продуктами: платформою підтримки навчання Moodle, веб-порталом на основі ТуроЗ, системою вебінарів на основі Blue Button, та виключає проблеми із ліцензуванням. Система має модульну структуру, що дозволяє проводити подальшу розробку та удосконалення. В проектуванні системи приймали участь співробітники підрозділів підвищення кваліфікації Національного технічного університету України «Київський Політехнічний Інститут». Це забезпечило відповідність системи потребам які існують в сучасних інститутах підвищення кваліфікації.

#### **Висновки**

Запропоновано систему підтримки навчального процесу підвищення кваліфікації, яка реалізовує модулі управління в сучасному веб-просторі навчання. Систему інтегровано в навчальний веб-простір, злагоджено взаємодію з іншими системами та платформами навчання, підключено до системи єдиного доступу користувачів.

Система встановлена та успішно працює протягом року в Науково Методичному Комплексі «Інститут Післядипломної Освіти» Національного Технічного Університету України «Київський Політехнічний Інститут». В системі зареєстровано близько 10000 користувачів.

#### **Література**

1. Валуйський, В.Н. Використання освітнього WEB-простору для навчання [Текст] / В.Н. Валуйський, М.В. Гончаренко, А.А. Павловський // Зб. наук. праць V міжнародної конференції "Стратегія якості в промисловості і освіті" 6-13 червня 2009 р., - Варна, Болгарія, ТОМ II, с.506-510 .

2. Галактика Управління ВУЗом / Інтернет-ресурс. – Режим доступу: [www/ URL: http://vuz.galaktika.ru](http://www.vuz.galaktika.ru). – Загол.с екрану.

3. ASoft Work&Education [Electronic resource] / Інтернет-ресурс. – Режим доступу: [www/ URL: http://www.asoftcrm.com/solution/crm\\_edu/index.php](http://www.asoftcrm.com/solution/crm_edu/index.php) – Загол.с екрану.

4. Open LDAP [Electronic resource] / Інтернет-ресурс. – Режим доступу: [www/ URL: http://www.openldap.org/](http://www.openldap.org/) – Загол.с екрану.

5. Single Sign-On [Electronic resource] / Інтернет-ресурс. – Режим доступу: [www/ URL: http://www.opengroup.org/security/sso/](http://www.opengroup.org/security/sso/). – Загол.с екрану.

6. Валуйский В.Н., Гончаренко М.В., Павловський А.А. «Разработка дистанционных курсов с использованием e-learning платформы Moodle» [Дистанційний курс] — Режим доступу: [www/ URL: http://moodle.udec.ntu-kpi.kiev.ua/moodle/course/view.php?id=25](http://moodle.udec.ntu-kpi.kiev.ua/moodle/course/view.php?id=25).

УДК 519.246.8

## ГРАНИЦЫ ПРИМЕНИМОСТИ КРИТЕРИЯ КЕЛЛИ ДЛЯ ДИНАМИЧЕСКОГО УПРАВЛЕНИЯ КАПИТАЛОМ В СЛУЧАЕ УСЕЧЕННОГО НОРМАЛЬНОГО РАСПРЕДЕЛЕНИЯ ДОХОДНОСТИ ТОРГОВОЙ СИСТЕМЫ

Гурьянова Т.В.

Донецкий национальный технический университет  
кафедра прикладной математики и информатики  
[gurianova\\_taya@ukr.net](mailto:gurianova_taya@ukr.net)**Аннотация**

*Гурьянова Т.В. Границы применимости критерия Келли для динамического управления капиталом в случае усеченного нормального распределения доходности торговой системы. Найдена область применения критерия Келли. Определен оптимальный алгоритм для определения доли капитала для реинвестирования, основанный на критерии Келли.*

**Введение.**

Любая торговая система (ТС) может быть охарактеризована одномерным законом распределения вероятностей ее выходных случайных величин  $\{x_i\}$  (выигрышей и проигрышей) на интервале стационарности их реализации  $\tau_{cm}$ . В большинстве практических случаев распределение доходностей подчиняется нормальному закону распределения, что вытекает из центральной предельной теоремы теории вероятностей (когда на систему действует одновременно большое количество разнонаправленных случайных факторов). В этом случае на интервале стационарности  $\tau_{cm}$  система характеризуется двумя числовыми характеристиками: математическим ожиданием  $M > 0$  и дисперсией  $\sigma^2$ , которые являются постоянными и не зависят от времени.

В то же время, для реальной торговой системы на значения  $\{x_i\}$  накладываются некоторые ограничения, которые в нашем исследовании сформулированы следующим образом: максимальный проигрыш системы не может быть больше 100%, максимальный выигрыш не может превышать значения 100%. Таким образом,  $x_i \in [-1; +1]$ ,  $i = 1, \dots, n$ .

Т.к. в рассматриваемой торговой системе возможен проигрыш всех вложенных в сделку средств (при  $x_k = -1$ ), то инвестору следует вкладывать в каждую совершаемую сделку только часть своего капитала  $f$ . Следовательно, после  $n$  сделок капитал игрока можно найти по формуле:

$$K_n = K_0 \cdot \prod_{i=1}^n (1 + x_i \cdot f), \quad (1)$$

где  $K_n$  - капитал игрока после  $n$  сделок,  $K_0$  - первоначальный капитал игрока,  $x_i$  - выигрыш (проигрыш) системы в  $i$ -й сделке,  $f$  - часть капитала, вложенная в каждую сделку (причем,  $0 \leq f \leq 1$ ). Тогда коэффициент приумножения капитала за  $n$  сделок:

$$\frac{K_n}{K_0} = \exp\left(n \ln\left(\frac{K_n}{K_0}\right)^{\frac{1}{n}}\right) = e^{n \ln G}, \quad (2)$$

где  $G$  - среднее геометрическое доходности или темп роста капитала (капитал экспоненциально растет, если  $G > 1$  и экспоненциально уменьшается при  $G < 1$ ). Дж. Келли в [1], показал, что для максимизации капитала необходимо максимизировать логарифм ожидаемой средней геометрической доходности  $G$ .

**Общая постановка проблемы.** В монографии [2] Р. Винсом был предложен параметрический метод определения оптимальной части капитала  $f$  при реинвестировании. В этом случае для нахождения  $f$  необходимо максимизировать логарифм темпа роста капитала при реинвестировании, что может быть выражено выражением:

$$\ln G|_{max} = \int_{-1}^1 \ln(1 + f \cdot x) \cdot w(x) \cdot dx, \quad (3)$$

где  $w(x)$  - заданная функция плотности вероятности закона распределения доходности системы. При анализе выражения (3) использовался метод Ньютона [3]. Для заданного закона распределения  $w(x)$  методом Симпсона [3] рассчитывалось значение интеграла (3), затем выбиралось начальное приближение значения  $f$  (как правило  $f = 0,5$ ), далее проводилось уточнение решения. Расчеты значений  $f$  проводилось с точностью до  $10^{-3}$ , поиск решения проводился на отрезке  $f \in [0; +1]$ . Численное моделирование проводилось в среде табличного процессора MS Excel.

Расчеты значений доли капитала для реинвестирования  $f$  проводилось для  $M \in [0,02; 1]$  с шагом 0,02,  $\sigma \in [0,02; 1]$  с шагом 0,02.

Из анализа полученных результатов следует, что существует достаточно широкая область значений параметров  $M$  и  $\sigma$ , для которых значение  $f$  почти не отличается от 1 (так, при любых значениях  $M$  и  $\sigma \leq 0,14$  значение  $f \geq 0,999$ ). В таких торговых системах применение критерия Келли приведет к разорению. Также, существует область значений параметров  $M$  и  $\sigma$ , для которых значение темпа роста капитала  $G < 1$ . В торговых системах, генерирующих временные ряды с такими параметрами, применение критерия Келли для нахождения доли капитала для реинвестирования приведет к разорению инвестора с вероятностью, равной 1.

Поэтому при проведении дальнейших исследований эти области исключались из рассмотрения, рассматривались торговые системы, генерирующие прибыли и убытки с параметрами  $M$  и  $\sigma$ , находящимися внутри области, изображенной на диаграмме (рис. 1).

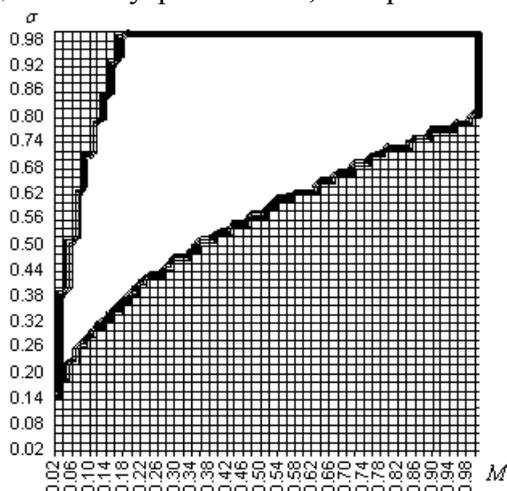


Рисунок 1. Область применения критерия Келли для случая нормального усеченного закона распределения доходности торговой системы.

В результате данных расчетов были найдены доли для реинвестирования  $f(M, \sigma)$  и соответствующие им темпы роста капитала  $G(M, \sigma)$  в зависимости от характеристик временного ряда, генерируемого торговой системой.

В реальности дело обстоит несколько иначе: при работе торговой системы характеристика  $M$  изменяется со временем и применение найденных значений  $f(M, \sigma)$  не является оптимальным.

Для имитации нестационарности торговой системы были смоделированы данные с постоянным значением  $\sigma$  и переменным  $M$ . В работе рассматривался случай плавного изменения  $M$  системы во времени. (рис. 2). Для анализа эффективности управления капиталом для нестационарного ряда использовался метод Монте-Карло.

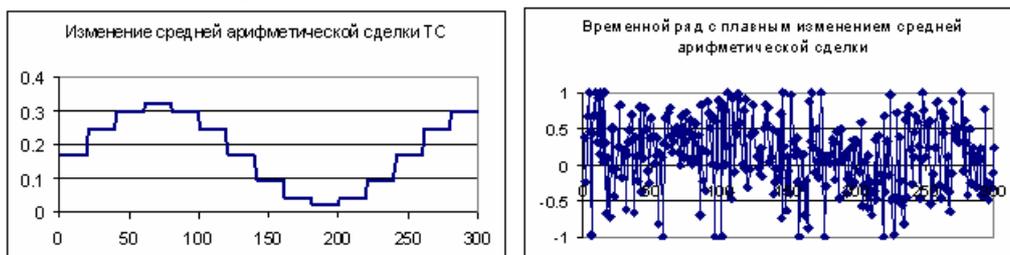


Рисунок 2. Пример временного ряда, сгенерированного торговой системой с плавно изменяющейся средней арифметической сделкой.

Было сгенерировано 80 временных рядов  $\{x_i\}$  с  $\sigma = 0,47$  и  $M \in [0,06; 0,32]$  по 300 сделок в каждом. Были протестированы следующие адаптивные алгоритмы динамического управления капиталом:

- алгоритм №1  $K_{/max} = \prod_{j=1}^{\tau} (1 + x_j \cdot f)$ ;

- алгоритм №2 [4]  $K_{/max} = \prod_{j=1}^{\tau} \left( 1 + \frac{-x_j}{x_k} \cdot f \right)$ ;

- алгоритм №3 [4], где  $m = 2$   $K_{/max} = \prod_{j=1}^{\tau} \left( 1 + \frac{-EMA[x_j]}{EMA[x_k]} \cdot F_2 \right)$ ;

- алгоритм №4 [4], где  $m = 5$   $F_3 = EMA[f^*_j] + \frac{2}{m+1} (f_j^* - EMA[f^*_{j-1}])$ .

Расчеты проводились следующим образом: в ряде  $\{x_i\}$ ,  $i = 1, \dots, 300$  первые 100 сделок считались историческими данными (т.е. во внимание при подсчете коэффициента умножения капитала  $K$  не принимались), а на сделках №101-300 проводился подсчет коэффициента умножения капитала  $K^1_{\tau}, K^2_{\tau}, K^3_{\tau}, K^4_{\tau}$  для каждого из предложенных алгоритмов для периода адаптации  $\tau = 10, 20, \dots, 50$ , данные заносились в таблицу. В каждой строке таблицы выбиралось  $K_{max}$  и данные каждой строки нормировались на этот коэффициент.

Рассмотрим коэффициенты, полученные для временного ряда №1 при  $\tau = 10$ :  $K^1_{10} = 0$ ,  $K^2_{10} = 296$ ,  $K^3_{10} = 334$ ,  $K^4_{10} = 1173$ , тогда  $K_{max} = 1173$  и  $K^1_{10}/K_{max} = 0$ ,  $K^2_{10}/K_{max} = 0,25$ ,  $K^3_{10}/K_{max} = 0,28$ ,  $K^4_{10}/K_{max} = 1$ . Выражение  $K^1_{10}/K_{max} = 0$  означает, что при применении алгоритма ДУК №1 при интервале адаптации  $\tau = 10$  капитал игрока упал до нуля;  $K^4_{10}/K_{max} = 1$  означает, что из всех алгоритмов ДУК алгоритм №4 дал наилучший коэффициент умножения капитала, т.е. этот алгоритм – наилучший для этой ТС на данном интервале адаптации; значения  $K^2_{10}/K_{max} = 0,25$ ,  $K^3_{10}/K_{max} = 0,28$  показывают, что на данном интервале адаптации  $\tau = 10$  для данной ТС алгоритмы №2, 3 не являются оптимальными и дают умножение капитала в, соответственно, 4 и 3,6 раза меньше, чем алгоритм №4.

Результаты анализа гистограммы распределения доходностей приведены в Таблице 1.

Таблица 1. Результаты анализа гистограммы распределения доходностей для периода адаптации  $\tau = 10$ .

$K/K_{max}$	Алгоритм №1	Алгоритм №2	Алгоритм №3	Алгоритм №4
$K/K_{max} = 0$	27	0	3	0
$0 < K/K_{max} \leq 0,5$	38	63	40	48
$0,5 < K/K_{max} \leq 0,99$	8	8	28	19
$K/K_{max} = 1$	7	9	9	13

При сравнении распределений показателя доходностей для разных алгоритмов установлено статистически значимое их различие ( $\chi^2=96,4$ ,  $p < 0,001$ ). Из проведенного анализа следует, что на периоде адаптации  $\tau = 10$  наибольшее количество разорений – 27 приносит алгоритм №1, наибольшее количество максимального умножения капитала – 13 – алгоритм №4.

Аналогичный анализ проведен для периодов адаптации  $\tau = 10, 20, \dots, 50$ . Результаты приведены в таблице 2.

Таблица 2. Результаты анализа гистограммы распределения доходностей при  $\tau = 10, 20, \dots, 50$ .

	Алгоритм №2					Алгоритм №3					Алгоритм №4				
	10	20	30	40	50	10	20	30	40	50	10	20	30	40	50
$K/K_{max} = 0$	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0
$0 < K/K_{max} \leq 0,5$	75	67	57	56	56	55	42	34	43	52	78	66	60	60	58
$0,5 < K/K_{max} \leq 0,99$	4	11	22	22	20	11	24	30	26	20	1	12	16	18	22
$K/K_{max} = 1$	1	2	1	2	4	11	14	16	11	8	1	2	4	2	0

Распределение значений доходностей для разных алгоритмов статистически значимо различается ( $\chi^2=203,6$ ,  $p < 0,001$ ). Из проведенного анализа может быть сделан вывод, что статистически наиболее оптимальный алгоритм ДУК – алгоритм №3 (однако его нецелесообразно применять при интервале адаптации  $\tau = 10$ ). Оптимальным интервалом адаптации при данных условиях ( $\sigma = 0,47$ ,  $M \in [0,06; 0,32]$ ,  $\tau_{cm} = 20$ ) является интервал  $\tau = 30$ .

### Выводы.

1. Проведенный анализ позволяет сделать вывод об общих закономерностях зависимостей  $f(M, \sigma)$  и  $G(M, \sigma)$  от параметров торговой системы и выбрать оптимальные параметры моделей торговых систем и оптимальные алгоритмы ДУК.

2. Наилучшим алгоритмом для нахождения оптимальной доли для реинвестирования является алгоритм №3.

### Литература

1. J. L. Kelly Jr. A New Interpretation of Information Rate. The Bell System Technical Journal, 35(4), 1956. – P. 917-926.
2. Винс Р. Математика управления капиталом. Методы анализа риска для трейдеров и портфельных менеджеров: Пер. с англ. – М.: Альпина Паблишер, 2001. – 400 с.
3. Бахвалов Н.С. Численные методы / Н.С. Бахвалов, Н.П. Жидков, Г.М. Кобельков. – 6-е изд. – М.: Бином, 2008. – 636 с.
4. Смирнов А.В., Гурьянова Т.В. Новое в динамическом управлении. Наукові праці Донецького національного технічного університету, серія «Інформатика, кібернетика та обчислювальна техніка», вып. 10 (153), Донецьк, ДонНТУ, 2009. – С. 230-233.

УДК 004.414.23

## КРИТЕРИЙ ОПТИМАЛЬНОСТИ ПРОТОКОЛА МАРШРУТИЗАЦИИ БЕСПРОВОДНЫХ СЕНСОРНЫХ СЕТЕЙ

**Кудр Латиф**

Донецкий национальный технический университет  
кафедра автоматизированных систем управления  
E-mail: multmediakar@mail.ru

### *Аннотация*

*Кудр Латиф. Критерий оптимальности протокола маршрутизации беспроводных сенсорных сетей. В настоящее время становятся все более популярным применение беспроводных сенсорных сетей в задачах автоматизированного управления и мониторинга. Самой популярной операционной системой в этой области является TinyOS в которой используется протокол маршрутизации СТР. Несмотря на большую распространенность, этот протокол имеет некоторые недостатки, поэтому задача оптимизации протоколов маршрутизации остается актуальной.*

### **Технологии беспроводных сенсорных сетей**

Беспроводная сенсорная сеть - это распределенная сеть, состоящая из автономных, удаленных друг от друга узлов. Узлы беспроводных сенсорных сетей - это миниатюрные устройства, содержащие процессор, оперативную и флэш-память, радиомодуль, АЦП, источник питания и датчики [1]. Набор используемых датчиков зависит от функций, выполняемых беспроводной сенсорной сетью, но в основном используются следующие типы датчиков: температурные, давления, влажности, освещения и др.

Узлы такой системы транслируют сообщения друг через друга, обеспечивая значительную площадь покрытия сетью при малой мощности передатчика. Таким образом, вся информация передается по сети до тех пор, пока не достигнет шлюза. Обычно шлюзом является такой же узел беспроводной сети, но подключенный к базовой станции.

Все узлы в беспроводных сенсорных сетях находятся под управлением специальных операционных систем. Как правило, эти операционные системы значительно проще обычных – это обусловлено ограниченностью ресурсов вычислительных устройств, на которых они работают.

TinyOS - это первая операционная система, разработанная специально для технологии беспроводных сенсорных сетей. TinyOS - это операционная система с открытым кодом. Благодаря компонентной структуре она обеспечивает минимальный размер кода. В ее состав входят драйверы сенсоров, сетевые протоколы и утилиты сбора информации. При разработке TinyOS основное внимание уделялось обеспечению малого энергопотребления и возможности использования для программирования языка с высоким уровнем абстракции.

На данный момент TinyOS является фактически стандартом для беспроводных сенсорных сетей, именно поэтому эта платформа была выбрана для реализации поставленных задач [2].

### **Протокол СТР**

СТР - это базовый протокол для большинства сенсорных сетей реализованных с использованием TinyOS. Беспроводные сенсорные сети под управление данного протокола образуют древовидную структуру, в которой шлюзы являются корнями, а сенсорные узлы - листьями [2, 3, 4].

Для генерации маршрутов узлы используют градиент маршрутизации - EТХ. Корень имеет значение EТХ равное 0. EТХ узла рассчитывается как сумма EТХ его родительского узла и

ЕТХ связи с ним. При выборе родительского узла предпочтение отдается тому, где значение ЕТХ меньше.

Несмотря на то, что СТР является базовым протоколом в технологии беспроводных сенсорных сетей, он имеет недостатки. Один из таких недостатков - это то, что протокол не учитывает загруженность узлов при ретрансляции пакетов. Таким образом, в сети возникает постоянная перегрузка некоторых узлов, что приводит к их быстрому выходу из строя, тем самым уменьшая время функционирования сети. Для определения путей оптимизации протокола ЕТХ рассмотрим детальнее принцип маршрутизации пакетов в нем.

### Принцип определения кратчайшего пути в СТР

Протокол СТР работает таким образом, что каждый узел обладает информацией о качестве связи с каждым из своих соседних узлов. А каждый из соседних узлов знает длину кратчайшего пути к базовой станции.

На рисунке 1 узел 5 имеет трех соседей: 3, 4 и 6. Качество связи с этими узлами:  $C_{53} = 2$ ,  $C_{54} = 3$ ,  $C_{56} = 6$ . Стоимость кратчайшего пути определяется как сумма всех оценок качества связей между узлами в этом пути. Например, если базовым узлом сети на рис. 1 является узел № 1, то  $D_1=0$ ,  $D_2=1$ . Для узла № 5 протокол СТР выбирает в качестве следующего промежуточного узла № 4, и общая стоимость передачи от узла 5 к узлу 1 равняется 5.

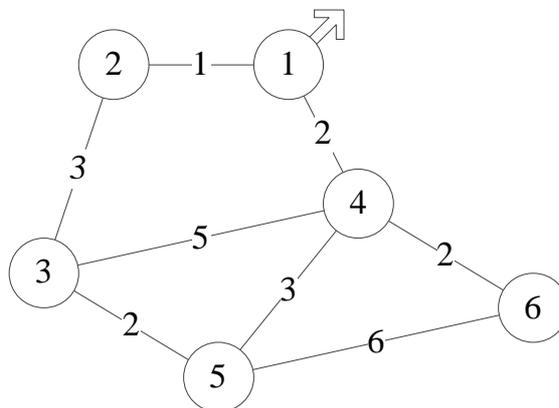


Рисунок 1 Пример структуры сети

$$D_5 = \min \{C_{53}+D_3, C_{54}+D_4, C_{56}+D_6\} = \min \{2+4, 3+2, 6+4\} = 5.$$

Рассмотрим принципы выбора ретранслирующего узла:

- каждый узел независимо определяет минимальную суммарную стоимость передачи;
- каждый узел периодически рассылает рассчитанную им минимальную суммарную стоимость своим соседним узлам;
- получая минимальную общую стоимость от соседних узлов, узел обновляет информацию о своей минимальной стоимости, если обнаруживается лучший путь и рассылает это новое значение соседям.

На этапе инициализации каждый узел устанавливает значение «неопределенная» для своей суммарной стоимости передачи к базовому узлу. Базовый узел устанавливает свою стоимость в 0. Когда соседи получают сообщение от базового узла, они обновляют свои стоимости передачи. Узлы, обновившие свое значение распространяют эту информацию дальше по сети. Шаг за шагом каждый узел определяет минимальное расстояние к базовому узлу. На рисунке 2 изображен конечный вид дерева маршрутизации.

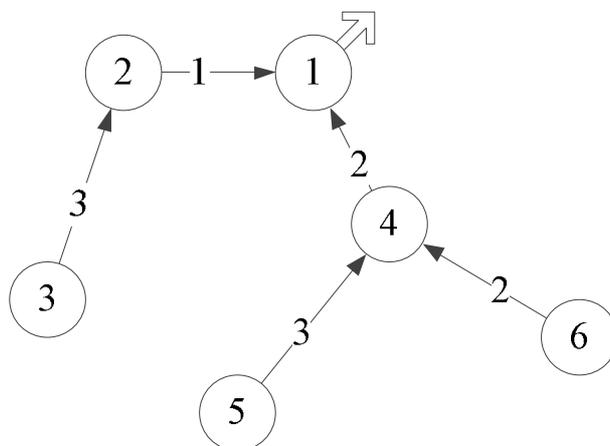


Рисунок 2 Конечный вид дерева маршрутизации

Каждый узел в СТР содержит таблицу соседей, в которой хранится ID узла и его стоимость передачи. В качестве метрики для стоимости передачи используется значение ETX (Expected Transmissions – ожидаемое количество передач).

Критерий оптимальности протокола маршрутизации в беспроводных сенсорных сетях должен учитывать следующие факторы. Во-первых – это среднее количество ретрансляций пакетов. Очевидно, что чем меньше количество ретрансляций, тем меньше расход энергии в целом по сети. Однако среднее не учитывает то, насколько сильно значения индивидуальных узлов могут отличаться от среднего значения для этих узлов. Поэтому критерий должен учитывать разброс, мерой которого может выступить, например дисперсия. В противном случае может оказаться так, что нагрузка на какой-нибудь отдельный узел может оказаться непредсказуемо высокой и этот узел откажет гораздо раньше, чем остальные узлы. Это может привести к потере связности сети и таким образом часть сети может оказаться неработоспособной. Поэтому мы предлагаем использовать наряду со средним значением, еще и меру разброса.

В простейшем случае величину разброса значений загруженности индивидуальных узлов можно оценивать по дисперсии. Однако дисперсия не дает представление о перегруженных узлах. Отклонения в обе стороны от среднего значения трактуются одинаково. Однако в нашем случае необходимо стремиться к тому, что в сети не будет узлов, нагрузка на которые превышает среднее значение. Незагруженные узлы не должны учитываться критерием. Поэтому мы предлагаем в качестве критерия использовать верхний квантиль, например квантиль порядка 0,95. Численное значение этого критерия равно значению нагрузки на узлы, которое не превышает только 5% самых загруженных из них. При этом нагрузка на узлы выражается в количестве пакетов ретранслируемых узлом за единицу времени.

Квантиль – это одна из числовых характеристик случайных величин, применяемая в математической статистике. Это такое число  $x_q$  что значения  $q$ -й части совокупности меньше или равны  $x_q$ . Например, квантиль 0,95 переменной - это такое значение ( $x_q$ ), что 95% значений переменной ниже этого значения.

Если функция распределения случайной величины  $X$  непрерывна, то квантилем порядка  $p$  называется такое значение случайной величины, при котором функция ее распределения принимает значение, равное  $p$ , т.е.

$$F(x_p) = P(X < x_p) = p \quad (1)$$

Для дискретной случайной величины распределение задано следующим образом:

Значения  $x$  случайной величины  $X$   $x_1, x_2, \dots, x_k$   
 Вероятности  $P(X = x)$   $p_1, p_2, \dots, p_k$   
 Для перечисленных  $k$  значений вероятности  $p$  решение  $x_p$  уравнения (1) неединственно, а именно,

$$F(x) = p_1 + p_2 + \dots + p_m$$

для всех  $x$  таких, что  $x_m < x < x_{m+1}$ . Т.е.  $x_p$  – любое число из интервала  $(x_m; x_{m+1}]$ . Для всех остальных  $p$  из промежутка  $(0;1)$  имеет место «скачок» со значения меньше  $p$  до значения больше  $p$ . А именно, если

$$p_1 + p_2 + \dots + p_m < p < p_1 + p_2 + \dots + p_m + p_{m+1},$$

то  $x_p = x_{m+1}$ .

Этот подход можно использовать для оценки оптимальности протокола маршрутизации, когда признаком предпочтительности является отсутствие узлов, нагрузка на которые по ретрансляции пакетов сильно превышает среднее значение.

Несмотря на то, что вопросам оптимизации протоколов маршрутизации в беспроводных сенсорных сетях посвящается множество работ зарубежных авторов [3, 4], задача является актуальной. Наш подход имеет преимущества, так как в нем при выборе узла ретрансляции учитывается загруженность узлов.

## Выводы

В работе предложен новый критерий оптимизации беспроводных сенсорных сетей – минимизация среднего количества ретранслируемых пакетов для наиболее загруженных узлов. Целевой функцией оптимизационной задачи является квантиль порядка 0,95 количества ретранслируемых пакетов каждым из узлов сети. Этот подход в отличие от использования математического ожидания и дисперсии имеет следующее преимущество: при оценке разброса значения количества ретранслируемых пакетов он учитывает только перегруженный узлы, в то время как дисперсия учитывает как перегруженные узлы, так и узлы с низкой нагрузкой.

## Литература

1. Интеллектуальные системы на базе сенсорных сетей. // Институт точной механики и вычислительной техники им. С.А. Лебедева РАН, 2009. [http://www-ipmce.ru/img/release/is\\_sensor.pdf](http://www-ipmce.ru/img/release/is_sensor.pdf)
2. Gnawali O., Fonseca R. et al. Collection tree protocol. Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems - SenSys '09, 1.
3. Vilajosana X., Morell A., J. Lopez Vicario et al. "Generalizing CTP: Does Probabilistic Routing in WSN Make Sense?" International Journal of Distributed Sensor Networks, vol. 2010, Article ID 542419, 8 pages, 2010.
4. Zheng, J. Simulate the routing path selection of mobile motes in CTP protocol. sensiv.org. Retrieved January 18, 2011, from [http://www.sensiv.org/twiki/pub/SensIV/-IntroductionToTossim/Simulate\\_the\\_routing\\_path\\_selection\\_of\\_mobile\\_motes\\_in\\_CTP\\_protocol.pdf](http://www.sensiv.org/twiki/pub/SensIV/-IntroductionToTossim/Simulate_the_routing_path_selection_of_mobile_motes_in_CTP_protocol.pdf).

УДК 004

## МЕТОДЫ СИНТЕЗА УСТРОЙСТВА УПРАВЛЕНИЯ С МНОЖЕСТВЕННЫМИ ИСТОЧНИКАМИ КОДОВ

**Лаврик А.С.**

Донецкий национальный технический университет  
кафедра компьютерной инженерии  
[alexandrlavrik@yandex.ru](mailto:alexandrlavrik@yandex.ru)

### ***Актуальность.***

В настоящее время для реализации цифровых устройств (ЦУ) широко используются программируемые логические интегральные схемы (ПЛИС) типа CPLD (Complex Programmable Logic Device) обладающих разнообразными характеристиками. Постоянно растущие требования к параметрам ЦУ делают необходимой разработку эффективных методов, приводящих к уменьшению схемы устройства при реализации в заданном базисе, в частности – схемы устройства управления [2, 3].

Устройство управления (УУ) является одним из центральных блоков любой цифровой системы, и во многом определяет характеристики системы в целом [1]. В настоящее время существует множество эффективных методов оптимизации аппаратных затрат в схемах УУ, реализуемых в виде автоматов с «жесткой» и «программируемой» логикой.

Устройство управления может быть реализовано как композиция автоматов с «жесткой» и «программируемой» логикой [3]. Такие УУ называются композиционными микропрограммными устройствами управления (КМУУ), преимуществом которых является минимально возможный объем управляющей памяти (УП) микрокоманд.

Современные ПЛИС с архитектурой CPLD состоят из множества макроячеек, объединённых матрицей переключений. Каждая макроячейка имеет структуру, подобную программируемым матрицам логики (ПМЛ) с широкой входной шиной, которые могут выполнять логическую функцию от большого числа переменных (несколько десятков) [2, 4, 5].

Новейшие тенденции в разработке ЦУ направлены на снижение потребляемой мощности, размеров и стоимости конечного устройства. Это порождает актуальную задачу уменьшения числа корпусов ПЛИС в схеме УУ. При её решении следует учитывать не только специфику алгоритма управления, но и особенности элементного базиса.

### ***Цель и задачи исследований.***

*Целью работы* является уменьшение аппаратных затрат в схеме КМУУ за счёт одновременного использования нескольких источников кодов классов псевдоэквивалентных ОЛЦ.

*Объект исследования* – синтез композиционных микропрограммных устройств управления с общей памятью (ОП) и разделением кодов (РК) на ПЛИС CPLD.

*Предмет исследования* – методы уменьшения аппаратных затрат в логической схеме КМУУ с ОП и КМУУ с РК на ПЛИС CPLD.

### ***Основные задачи исследований:***

1. Анализ особенностей элементного базиса ПЛИС, существующих структур и методик синтеза КМУУ, ориентированных на реализацию в этом базисе.

2. Разработка структур и модификация методов уменьшения аппаратных затрат в схеме КМУУ с общей памятью и КМУУ с разделением кодов ориентированная на эффективное использование особенностей базиса однородных ПЛИС.
3. Разработка алгоритма выбора модели КМУУ с наименьшими аппаратными затратами для заданной ГСА, основанного на результатах анализа характеристик ГСА и базиса реализации схемы.
4. Исследование логических схем, порождаемых разработанными моделями КМУУ, с целью определения областей их эффективного применения.

**Научная новизна полученных результатов** определяется следующими положениями:

1. Разработаны новые структуры и методы синтеза КМУУ с множественными источниками кодов на основе КМУУ с общей адресацией и КМУУ с разделением кодов, позволяющие уменьшить аппаратные затраты на реализацию схемы адресации и преобразователя кода адреса при реализации на CPLD.
2. Впервые применен подход к использованию свободных выходов ПЗУ для размещения части преобразователя кодов в структурах КМУУ с общей памятью и КМУУ с разделением кодов при реализации на CPLD.
3. Получены аналитические зависимости числа задействованных макроячеек для реализации узлов КМУУ от различных параметров граф-схем алгоритмов.

**Практическое значение полученных результатов** состоит в разработке новых структур КМУУ в базисе программируемых логических схем с архитектурой CPLD, приводящих к уменьшению числа макроячеек – как минимум в 1,2 раза по сравнению с традиционными структурами при реализации на CPLD. А также в получении аналитических зависимостей аппаратных затрат в схемах КМУУ от параметров исходной граф-схемы алгоритма.

#### **Методы исследования.**

Необходимые для проведения исследований данные были получены с помощью программного комплекса состоящего из следующих частей (рис. 1).

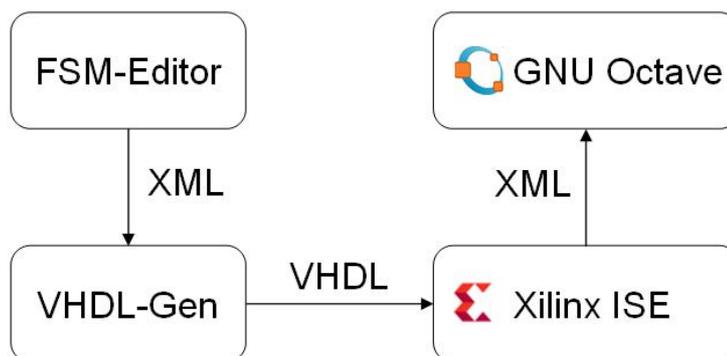


Рисунок 1. Структура разработанного программного комплекса для проведения исследований.

В состав системы входят:

1. FSM-Editor – специализированное средство позволяющее создавать граф-схемы алгоритма (ГСА) путём их графического ввода или автоматической пакетной генерации согласно заданным параметрам. Далее, описание алгоритма можно сохранить в формате XML.

2. VHDL-Generator – специализированное средство предназначенное для получения VHDL кода необходимого устройства управления согласно выбранной структуре и описанию алгоритма в формате XML.
3. Xilinx ISE – средство разработки, бесплатно предоставляемое компанией Xilinx. Позволяет на основе VHDL описания устройства управления получить оценочные характеристики аппаратных затрат при реализации алгоритма на ПЛИС CPLD.
4. GNU Octave – бесплатный аналог MathLab. Используется для анализа данных, полученных от Xilinx ISE и построения графиков.

### **Обзор полученных результатов.**

В результате работы программного комплекса были получены некоторые зависимости, отражающие эффективность некоторых из предложенных структур (рис. 2а, 2б). На рисунках отражена поверхность, определяющая при каких условиях достигается минимум аппаратных затрат. Поверхность окрашена цветом той структуры, для которой достигается минимум в каждой её точке.

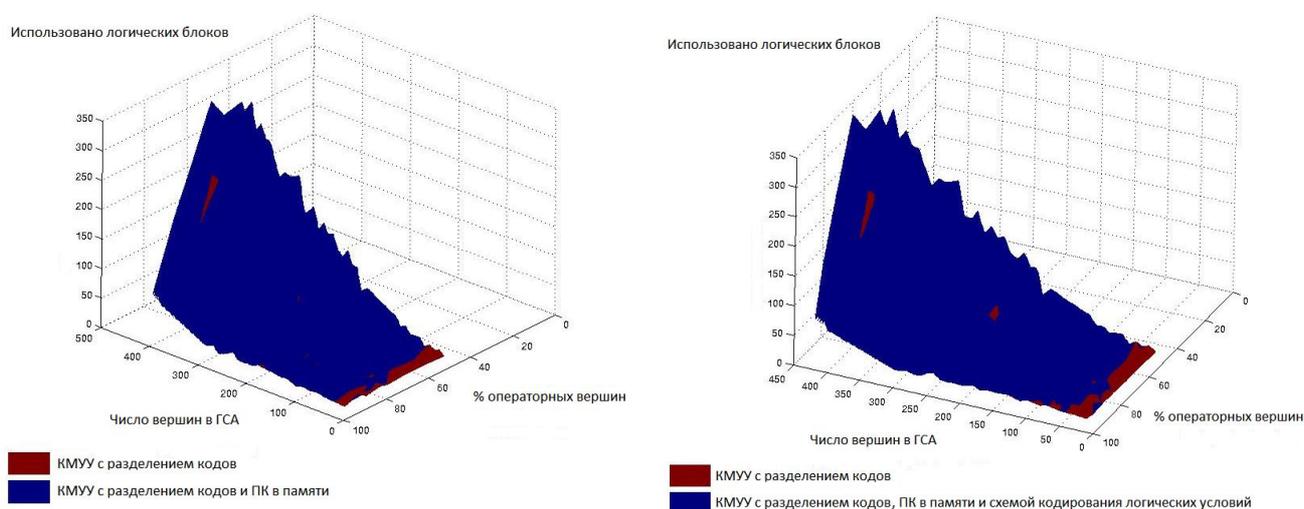


Рисунок 2. Зависимость эффективности применения некоторых структур от числа вершин в ГСА и доли операторных вершин в них.

### **Литература**

1. *Baranov S.* Logic Synthesis for Control Automata. – Kluwer Academic Publishers, 1994. – 312 pp.
2. *Соловьев В.В.* Проектирование цифровых схем на основе программируемых логических интегральных схем. – М.: Горячая линия-ТЕЛЕКОМ, 2001. – 636 с.
3. *Barkalov A., Titarenko L.* Logic Synthesis for Compositional Microprogram Control Units. – Berlin: Springer, 2008. – 272 pp.
4. Электронный ресурс. Altera devices overview.  
[http://www.altera.com/products/devices/common/dev-family\\_overview.html](http://www.altera.com/products/devices/common/dev-family_overview.html).
5. Электронный ресурс. Xilinx CPLDs  
[http://www.xilinx.com/products/silicon\\_solutions/cplds/index.htm](http://www.xilinx.com/products/silicon_solutions/cplds/index.htm).

УДК 004.8:330.4:658

**ДОСЛІДЖЕННЯ ПІДХОДІВ ДО ОЦІНЮВАННЯ РИЗИКІВ У МОДЕЛЯХ  
ОПТИМІЗАЦІЇ ІНВЕСТИЦІЙНОГО ПОРТФЕЛЯ****Льовкін В.М.**Запорізький національний технічний університет  
Кафедра програмних засобів  
vliovkin@gmail.com**Анотація**

*Льовкін В.М. Дослідження підходів до оцінювання ризиків у моделях оптимізації інвестиційного портфеля. Розглянуто проблему оцінювання ризиків інвестиційних портфелів. Досліджено сучасні моделі оптимізації інвестиційного портфеля щодо підходів до оцінювання ризиків, які використовуються в них. Визначено основні їх переваги та недоліки, сформульовано умови застосування. Розроблено інформаційну технологію оптимізації інвестиційного портфеля.*

**Вступ**

Управління ризиками посідає особливе серед підтримки інвестиційних рішень. Причини, що зумовлюють необхідність економічних інвестицій: оновлення матеріально-технічної бази, нарощування обсягів виробничої діяльності, освоєння нових видів діяльності. На прийняття рішення щодо інвестування впливають такі чинники: вид інвестицій; вартість інвестиційного проекту; кількість раціональних варіантів; обмеженість фінансових ресурсів; ризик прийняття рішення; інші умови та обставини [1].

Відомо, що чим більша спеціалізація підприємства, тим чіткіший ритм виробництва, вища його рентабельність, але при цьому гірша адаптивність даної економічної системи до змін зовнішнього економічного середовища. Так, на ринку цінних паперів (ЦП) інвестор не повинен вкладати гроші в ЦП лише одного виду, необхідна диверсифікація вкладень, бо в протилежному випадку він приречений або на низьку ефективність (норму прибутку), або на надто високий ступінь ризику [2].

Диверсифікація – процес розподілу інвестованих коштів між різними об'єктами вкладання, які безпосередньо не зв'язані один з одним. Такий розподіл інвестицій знижує ризик, забезпечує більшу стійкість доходів за будь-яких коливань дивідендів і ринкових цін на ЦП. На принципі диверсифікації базується діяльність інвестиційних фондів, які продають клієнтам свої акції, а одержані кошти вкладають у різноманітні ЦП, які обертаються на фондовому ринку й дають стійкий середній прибуток.

Диверсифікація дозволяє уникнути частини ризику при розподілі капіталу між різноманітними видами діяльності. Так, придбання інвестором акцій різних акціонерних товариств замість акцій одного товариства збільшує ймовірність одержання ним середнього доходу майже втричі й відповідно в три рази знижує рівень ризику.

**Постановка завдання**

Під час вибору інвестиційного портфеля (ІП) важливо правильно оцінити майбутні ризики, адже від обраних фінансових інструментів на пряму залежить дохідність затрачених ресурсів. Невірна оцінка ризиків може призвести до зменшення прибутку, який можна було б отримати, або в крайньому випадку збитків. Тому інвестору важливо обрати той підхід, який би зважаючи на конкретні умови застосування, дозволив найбільш ефективним шляхом вкласти наявні кошти в інвестиції. Виходячи з цього, **актуальною проблемою** є аналіз існуючих моделей оптимізації ІП щодо засобів оцінювання ризиків у них.

Завданням даної роботи є дослідження підходів до оцінювання ризиків у різних моделях оптимізації ІП, визначення недоліків та переваг, умов застосування та рекомендацій для застосування того чи іншого підходу.

### Традиційні підходи до оптимізації портфеля

Класична методика оптимізації ІП, модель Г.Марковіца [3], за міру ризику приймає стандартне відхилення величини дохідності: чим більше значення даного показника, тим більш ризикованим буде портфель. У своїх дослідженнях Марковіц виходив з того, що значення дохідності ЦП – випадкові величини, розподілені за нормальним законом. Інвестор повинен вибрати портфель, який забезпечує максимальну дохідність при прийнятному рівні ризику або мінімальний ризик при заданому рівні очікуваної дохідності. Ризик портфеля вимірюється не тільки індивідуальним ризиком кожної окремої ЦП портфеля, але і тим, що існує ризик впливу змін щорічних величин дохідності однієї акції на зміни дохідності інших акцій, що входять до складу ІП. За міру взаємозалежності двох випадкових величин використовують коваріацію та коефіцієнт кореляції. Якщо розглядаються величини дохідності ЦП за минулі періоди, то коваріація розраховується за наступною формулою:

$$\delta_{i,j} = \sum_{t=1}^N [r_{i,t} - E(r_i)] \times [r_{j,t} - E(r_j)] / (N - 1), \quad (1)$$

де  $\delta_{i,j}$  – коваріація між величинами дохідності ЦП  $i$  та ЦП  $j$ ;

$r_{i,t}$  та  $r_{j,t}$  – дохідність ЦП  $i$  та ЦП  $j$  в момент часу  $t$ ;

$E(r_i), E(r_j)$  – очікувана дохідність ЦП  $i$  та ЦП  $j$ ;

$N$  – загальна кількість періодів спостереження.

Задача оптимізації структури відповідного портфеля досягненням заданої дохідності з мінімальним ризиком називається задачею Марковіца і має наступний вигляд:

$$\delta_n^2 = \sum_{i=1}^n w_i^2 \delta_i^2 + \sum_i \sum_j w_i w_j P_{i,j} \delta_i \delta_j \rightarrow \min_w \quad (2)$$

$$E(r) = \sum_i w_i E(r_i) \quad (3)$$

$$\sum_{i=0}^n w_i = 1 \quad (4)$$

Головними недоліками класичного підходу є вимога щодо нормального розподілу значень дохідності, можливість розгляду тільки конкретних значень дохідності та необхідність встановлення ймовірності їх досягнення. Ризиком вважаються всі відхилення від запланованих показників, як у меншу, так і в більшу сторону.

У. Шарп запропонував новий метод побудови границі ефективних портфелів – одноіндексну модель Шарпа [4], яка дозволяє значно скоротити обсяг необхідних обчислень. Докладніше принципи даної моделі розглядалися в [5]. У моделі Шарпа не ліквідовано наведені в моделі Марковіца недоліки, проте зменшено обсяг обчислень за допомогою використання лінійної регресійної моделі.

### Методологія Value-at-Risk для оптимізації портфеля

Методологія Value-at-Risk (VaR) дозволяє оцінити величину максимально можливих збитків на визначеному горизонті планування зі встановленим рівнем ймовірності. Інвестор розглядає ризик, пов'язаний з несприятливими ситуаціями, як тільки несприятливі відхилення від очікуваних значень. У даній методології запропоновано за міру ризику використовувати

семіваріацію [1], яка для величини  $X$  обчислюється наступним чином:

$$SV(X) = \sum_{j=1}^n \alpha_j p_j (x_j - M(X))^2, \quad (5)$$

де  $n$  – кількість значень випадкової величини  $X$ ;

$x_j$  – значення випадкової величини,  $j = 1, \dots, n$ ;

$p_j$  – відповідні ймовірності;

$M(X)$  – математичне очікування випадкової величини  $X$ ;

$\alpha_j$  – індикатор несприятливих відхилень, що дорівнює 0 у випадку сприятливого відхилення і 1 – у протилежному.

Однак, одним із недоліків моделі Марковіца та класичної методології VaR є припущення про нормальність розподілу та наявність симетрії в розподілі. На практиці більшість економічних показників асиметричні. Тому в [6] за базу при розрахунках пропонується замість математичного очікування використати моду або медіану.

Таким чином, методологія VaR та розглянуті її модифікації дозволяють усунути такі недоліки моделі Марковіца: припущення про нормальність розподілу досліджуваної величини та врахування як негативних, так і позитивних відхилень значень досліджуваної величини під час обчислення міри ризику.

#### **Підхід до оптимізації портфеля на основі теорії нечітких множин**

В нечіткому підході на основі теорії можливостей [7] дохідність  $i$ -ого ЦП представляється у вигляді трикутного нечіткого числа

$$r_i = (r_{1i}; \bar{r}_i; r_{2i}), \quad (6)$$

де  $r_{1i}$  – нижня межа дохідності  $i$ -ого ЦП;

$\bar{r}_i$  – очікувана дохідність  $i$ -ого ЦП;

$r_{2i}$  – верхня межа дохідності  $i$ -ого ЦП.

Дохідність портфеля визначається наступним чином:

$$r = (r_{\min}; \bar{r}; r_{\max}) = \left( \sum_{i=1}^N w_i r_{1i}; \bar{r}; \sum_{i=1}^N w_i r_{2i} \right). \quad (7)$$

Критичний рівень дохідності портфеля на момент часу  $T$  представляється як  $r^* = (r_1^*; \bar{r}^*; r_2^*)$ . Тоді взаємне співвідношення функцій приналежності  $r_i$  та критеріального значення  $r^*$  дозволяє розрахувати площу фігури, яка утворюється в результаті перетину цих функцій. Ступінь ризику неефективності – геометрична ймовірність потрапляння точки  $(r, r^*)$  у зону неефективного розподілу капіталу:

$$\varphi(\alpha) = \frac{S_\alpha}{(r_2^* - r_1^*)(r_2 - r_1)} \quad (8)$$

Докладніше даний підхід до оцінювання ризиків було розглянуто авторами в [5]. Застосування даного підходу розглядалося в [8-9].

За умов даного підходу необхідно розв'язати наступну задачу оптимізації:

$$r = \sum_{i=1}^N w_i r_i \rightarrow \max; \quad (9)$$

$$\sum_{i=1}^N w_i r_{li} > r^* ; \quad (10)$$

$$\sum_{i=1}^N w_i = 1, w_i \geq 0; i = \overline{1, N}. \quad (11)$$

У даній моделі ліквідовано недоліки підходів на основі теорії ймовірності. Знято вимоги щодо нормального розподілу значень дохідності, розглядається не можливість досягнення окремих значень показника дохідності, а цілий інтервал значень, ймовірність досягнення яких не потрібно визначати наперед. Окрім того на відміну від моделі Марковіца при нечіткому підході за ризик приймаються тільки ті значення дохідності, які виявились менше запланованого.

### Короткострокова стратегія оптимізації портфеля

У підході до управління портфелем ЦП на основі D-оцінок Руссмана [10] ризик розглядається як максимум важкості досягнення цілі даного портфеля. Даний підхід до визначення ризику не обмежує область застосування методу, бо важкість досягнення мети еквівалентна ризику, як мірі загрози втрати керованості та еквівалентна вартості (витратам) на збереження керованості або вартості можливості досягнення мети. При цьому задача оптимізації полягає в формуванні портфеля таким чином, щоб ризик під час руху системи вздовж деякої прогнозованої траєкторії  $A = f(t)$  був мінімальним.

Важкість досягнення мети в деякий момент часу  $t$  ( $0 < t < t_{pl}$ ) залежить від положення системи в цей момент, а також від мінімальної та максимальної можливих швидкостей руху розглядаємої системи. Ризик як важкість досягнення мети визначається наступним чином:

$$d(t) = \max\{d_1(t), d_2(t)\}, \quad (12)$$

де  $d(t) \in [0,1]$ .

$$d_1 = \frac{-V_{\min}(V_{\max}t - f(t))(1 + V_{\min}t - V_{\min} - f(t))}{(f(t) - V_{\min}t)(V_{\max} - V_{\min}V_{\max} + V_{\min}V_{\max}t - k_1f(t))}, \quad (13)$$

$$d_2 = \frac{(V_{\max}t - f(t))(1 + V_{\min}t - V_{\min} - f(t))}{(1-t)(V_{\max} - 1)(V_{\max} - V_{\min})}, \quad (14)$$

де  $V_{\min}$  – мінімальна швидкість руху об'єкта до мети, тобто мінімальна швидкість збільшення вартості портфеля (при цьому дана швидкість може бути від'ємною);

$V_{\max}$  – максимальна швидкість збільшення вартості портфеля;

$f(t)$  – довільна функція, яка відбиває траєкторію руху системи до мети.

Даний підхід може бути використано в умовах нестабільної ситуації на ринку акцій, яка спостерігається на українському та російському ринках. Традиційні підходи розраховані на довгострокове інвестування, бо для оцінок ризику та дохідності вони використовують усереднені за великим проміжком часу характеристики. Підхід на основі D-оцінок Руссмана може бути використано також і для довгострокових стратегій, але таких, що передбачають частий перегляд портфеля.

Розроблено інформаційну технологію, яка дозволяє на основі досліджуваних моделей та методів виконувати як довгострокову, так і короткострокову оптимізацію ПП, використовуючи проаналізовані в роботі підходи до оцінювання ризиків ПП. Використання даної системи

дозволить інвестору приймати рішення щодо інвестування в залежності від умов, у яких він діє, на основі наявного інструментарію.

### **Висновки**

Класична модель оптимізації П, модель Марковіца, має ряд припущень, які не відповідають реальній ситуації на ринку ЦП.

Методологія Value-at-Risk дозволяє враховувати в якості міри ризику тільки негативні відхилення значень досліджуваного показника. Окрім того, модифікована методологія Value-at-Risk, в якій за міру ризику прийнято медіанну семіваріацію, дозволяє також ліквідувати припущення моделі Марковіца про нормальний закон розподілу значень дохідності.

В умовах кризових явищ, а також нестабільності ринку, характерного для України, дані моделі виявляються неефективними. Якщо досліджуваний ринок неоднорідний і необхідно розглянути весь інтервал можливих значень дохідності, потрібно застосувати нечіткий підхід на основі теорії можливостей. Якщо наявні дані про зміну ринка недостатньо точні, то для короткострокової оптимізації варто застосовувати підхід до управління портфелем на основі D-оцінок Руссмана, який розглядає ризик інвестиційного портфеля як міру загрози втрати керованості системи.

### **Література**

1. Вітлінський, В.В. Ризикологія в економіці та підприємстві : Монографія [Текст] / В.В. Вітлінський, Г.І. Великоіваненко – К. : КНЕУ, 2004. – 480 с.
2. Вітлінський, В.В. Аналіз, оцінка і моделювання економічного ризику [Текст] / В.В. Вітлінський. – Київ : ДЕМІУР, 1996. – 212 с.
3. Markowitz, H.M. Portfolio Selection [Текст] / H.M. Markowitz // Journal of Finance. – 1952. – № 7. – Pp. 77-91.
4. Sharpe, W.F. A Simplified Model for Portfolio Analysis [Текст] / W.F. Sharpe // Management Science. – 1963. – Vol. 9, № 2. – Pp. 277-293.
5. Дубровін В.І. Оцінювання ризиків інвестиційного портфеля [Текст] / В.І. Дубровін, В.М. Льовкін // Радіоелектроніка. Інформатика. Управління. – 2010. – №1 (22). – С. 51-55.
6. Скіцько, В.І. Оцінка ризику методом Value-at-Risk [Текст] / В.І. Скіцько // Економіка : проблеми теорії та практики : Зб. наук. праць. – Дніпропетровськ : ДНУ, 2005. – № 202. – С. 158-165.
7. Недосекин, А.О. Методологические основы моделирования финансовой деятельности с использованием нечетко-множественных описаний : дис. на соискание уч. степени докт. экон. наук: спец. 08.00.13 “Математические и инструментальные методы экономики” [Текст] / А.О. Недосекин. – Санкт-Петербургский государственный университет экономики и финансов. – СПб., 2003. – 280 с.
8. Льовкін, В.М. Оцінювання ризиків при управлінні інвестиційними проектами [Текст] / В.М. Льовкін // 14-й міжнародний молодіжний форум «Радіоелектроніка і молодь в ХХІ ст.» : Зб. матеріалів форуму. – Харків : ХНУРЕ, 2010. – ч.2. – С.224.
9. Дубровін, В.І. Оцінювання ризиків в процесі оптимізації інвестиційного портфелю [Текст] / В.І. Дубровін, В.М. Льовкін // Тиждень науки – 2010 : зб. тез доп. щоріч. наук.-практ. конф. викладачів, науковців, молодих учених, аспірантів, студентів ЗНТУ (Запоріжжя, 12–16 квіт. 2010 р.) ; в 4 т., т. 2 ; відп. ред. Ю. М. Внуков. – Запоріжжя : ЗНТУ, 2010. – С. 126-127.
10. Иванова, К.Г. Управление портфелем ценных бумаг на основе D-оценок Руссмана и нейросетевого моделирования : дисс. на соискание уч. степени канд. экон. наук : спец. 08.00.13 “Математические и инструментальные методы экономики” [Текст] / К.Г. Иванова. – Воронежский государственный университет. – Воронеж, 2009. – 129 с.

УДК- 004.93`1 + 004.896

## ПОСТРОЕНИЕ НЕЧЕТКОГО КЛАССИФИКАТОРА В ЗАДАЧЕ КОНТРОЛЯ КАЧЕСТВА НЕФТЕПРОДУКТОВ

**Максимова А.Ю.**

Институт прикладной математики и механики НАН Украины

Донецк

E-mail: lunaplus@mail.ru

### *Аннотация*

*Максимова А.Ю. Построение нечеткого классификатора в задаче контроля качества нефтепродуктов. Решается задача классификации жидких нефтепродуктов. Для решения задачи используется алгоритм распознавания образов на базе нечетких портретов. Результат работы алгоритма представляется в виде нечеткого множества. Для повышения качества распознавания было выполнено разбиение на кластеры некоторых классов образов, что позволило улучшить качество распознавания на 2% по сравнению с исходным алгоритмом (92% верно распознанных объектов).*

### **Введение**

В различных областях человеческой деятельности возникают задачи, сводящиеся к классу задач распознавания образов или классификации. Развитие компьютерных технологий позволило накапливать большие объемы информации, требующие обработки и преобразования с целью извлечения из них необходимых данных и знаний. В связи с этим активно разрабатываются методы обработки данных, и, в частности, методы распознавания образов. По [1] целью разведочного анализа данных (exploratory data analysis) является выявление структуры данных и описание ее статистическими моделями. Однако этот подход требует достаточно жестких модельных ограничений, вытекающих из аксиоматики теории вероятностей. Выход за рамки статистических моделей привел к формированию более общего направления интеллектуального анализа данных, в котором используются методы нечеткой логики, эволюционные и генетические алгоритмы, иммунные системы и др., а также, гибридные методы, которые в совокупности получили название «мягкие вычисления» (soft computing).

На практике исходные данные зачастую обладают сложно формализуемой, неоднородной структурой с заведомо пересекающимися классами образов. Другой проблемой является неполнота обучающей выборки и ее сильная зашумленность. В дискриминантном анализе в случаях неопределенности ответа о принадлежности образа классу образов ответ может быть получен в виде вероятности принадлежности образа каждому из классов образов. Однако описанные выше особенности данных во многих случаях не позволяют построить адекватные вероятностно-статистические модели, что обуславливает создание эмпирических методов и подходов, при разработке которых удобными становятся методы и модели из области мягких вычислений.

**Целью работы** является решение прикладной задачи классификации жидких нефтепродуктов (ЖНП), в частности бензинов. Эта задача возникает при контроле их качества по результатам лабораторных измерений некоторых их физико-химических характеристик. Для ее решения предлагается алгоритм распознавания образов на базе нечетких портретов, в основу которого положена идея построения интегральных характеристик классов образов в виде нечетких множеств – значений лингвистических переменных, описывающих классы образов по признакам, а принятие решения о

принадлежности рассматриваемого образца к определенному виду выполняется методами логического нечеткого вывода.

### **Особенности рассматриваемой задачи классификации жидких нефтепродуктов**

Как отмечалось выше, в работе предлагается метод для решения задачи контроля качества нефтепродуктов, которая сводится к задаче распознавания образов. Ее суть заключается в определении производителя и вида образца ЖНП в лабораторных условиях. В лаборатории контроля качества нефтепродуктов накапливается информация об образцах топлива, поступающих на экспертизу от разных производителей и потребителей. По каждому образцу разными методами определяется ряд показателей. Такими показателями являются октановое число, содержание ароматических веществ, в том числе олефинов, ароматических бензолов, содержание серы и др. Накопленная информация по всем образцам используется в качестве эталонной для определения производителя рассматриваемого образца. Отсутствие универсальных подходов к измерению параметров ЖНП и неоднородный характер накопленной информации сдерживают развитие методов и приборов для автоматизации процессов управления в лабораториях контроля качества. Существующие разработки обладают рядом недостатков и ограничений в использовании. Например, в работе [2] предлагается нейросетевой импедансный метод идентификации ЖНП, работающий с данными, полученными с импедансометрического датчика. Эти данные обрабатываются нейросетевым алгоритмом. Однако алгоритм ориентирован на уникальный прибор, предложенный автором указанного метода.

В качестве класса образов в данной задаче рассматривается топливо определенного вида от определенного производителя. Основной особенностью исходных данных является полное или частичное совпадение некоторых классов образов. Это связано с технологическими особенностями производства ЖНП, когда разные производители реализуют топливо одного вида и с одинаковыми характеристиками. В ситуациях, когда не удается однозначно определить, к какому классу образов относиться рассматриваемый образец, важно получить ответ о степени схожести образца на каждый из предполагаемых классов образов, а, иногда, определить, что это точно не образец определенного класса образов. В связи с этим удобно ответ получать в виде нечеткого множества, элементы которого - классы образов.

### **Постановка задачи**

Формально постановку прикладной задачи распознавания нефтепродуктов можно рассматривать как нечеткую модификацию задачи классификации в многомерном пространстве. Пусть дана обучающая выборка  $Y = \{(x^{(i)}, v^{(i)}), x^{(i)} \in X, v^{(i)} \in V, i = 1, \dots, n\}$ , где  $x^{(i)} \in X \subset R^m$  векторы  $n$ -мерного пространства – набор информативных признаков,  $V = \{v_i\}$ ,  $i = 1, \dots, k$ ,  $v_i \in N$  - множество номеров классов образов. Пары  $(x^{(i)}, v^{(i)})$  определяют, представителем какого класса образов  $v^{(i)}$  является образ  $x^{(i)}$ . В общем случае, необходимо определить степень принадлежности образца  $\bar{x}$  рассматриваемым классам, т. е. построить нечеткое множество  $\tilde{y}(\bar{x}) = \sum_{i=1}^k \mu_i(\bar{x}) / v_i$ . где  $\mu_i(\bar{x})$  - степень принадлежности образа классу  $v_i$ .

### **Алгоритм распознавания по нечетким портретам классов образов**

Для решения рассматриваемой задачи классификации нефтепродуктов используется предложенный в [3] алгоритм распознавания. Основная идея алгоритма заключается в представлении исходной информации о классах в виде их нечетких портретов, которые формируются в результате анализа обучающей выборки. Такие портреты описываются совокупностью лингвистических переменных, соответствующих информативным признакам. Терм-множества этих лингвистических переменных описывают значение признака для

каждого из классов образов и строятся в результате анализа частоты встречаемости значений признака в каждом классе образов. По построенным нечетким портретам формируется база знаний нечетких продукций, а решение принимается алгоритмом нечеткого вывода. Результат работы алгоритма представляется нечетким множеством.

В алгоритме берутся в рассмотрение признаки с низкой попарной корреляцией, поэтому данные по каждому показателю рассматриваются независимо друг от друга. Первоначальный выбор информативных признаков в задаче классификации ЖНП был осуществлен экспертом.

Рассмотрим более подробно алгоритм построения нечетких портретов. Для каждого информативного признака  $P_i$ ,  $i=1, \dots, m$  определим лингвистическую переменную  $L_i$ :  $L_i = \{eiy(P_i), T_i, U_i, G, M\}$ , где  $U_i$  - область определения значений признака  $P_i$ ,  $T_i = \{(\mu_{i,j}(x)), j=1, \dots, k\}$  - терм-множества лингвистической переменной,  $\mu_{ij}(pr_i \bar{x}) \in [0,1]$  - функция принадлежности, определяющая степень уверенности, с которой образ  $\bar{x}$  относится к классу образов  $v_j$ , причем  $pr_i \bar{x}$  определяет значение признака  $P_i$  для образа  $\bar{x}$ . Синтаксическое правило  $G$ , порождающее названия переменных, в данном случае тривиально, т.к. все термы атомарные, и заключается в присвоении функции принадлежности имени класса, который она представляет. Семантическое правило  $M$  представлено в виде алгоритма формирования функций принадлежности, который основан на концепции скользящего окна и является расширением подхода, используемого при построении гистограмм в статистике. Вид функций принадлежности зависит от выбора коэффициентов  $\alpha$  и  $\beta$ , которые фактически определяют ширину скользящего окна и шаг скольжения и являются настраиваемыми параметрами алгоритма распознавания.

Нечетким портретом первого порядка  $S_j$  класса  $v_j$  определяется как совокупность значений лингвистических переменных, соответствующих классу  $v_j$ :  $S_j = \{\mu_{ji}\}, i=1, \dots, n$ . Для каждого нечеткого портрета  $S_j$  строится правило нечеткого вывода по определенным ранее лингвистическим переменным.

ПРАВИЛО « $S_1$ »:

ЕСЛИ « $L_1$  есть  $v_1$ » И... И  $L_i$  есть  $v_1$  И ... И  $L_m$  апои  $v_1$  ТО  $Y$  есть  $(v_1 / \mu_{v_1})$ ;

...

ПРАВИЛО « $S_k$ »:

ЕСЛИ « $L_1$  есть  $v_k$ » И... И  $L_i$  есть  $v_k$  И ... И  $L_m$  апои  $v_k$  ТО  $Y$  есть  $(v_k / \mu_{v_k})$ .

Принятие решения осуществляется на основе механизма нечеткого вывода. Следует отметить, что в качестве операции «И» на этапе агрегирования используется  $m$ -местная функция  $f(a_1, a_2, \dots, a_m) = \log_2((a_1 + 1)(a_2 + 1) \dots (a_m + 1)) / m$ ,  $a_i \in (0,1]$ . Результатом работы алгоритма является нечеткое множество  $\tilde{y}$ .

Для решения задачи классификации нефтепродуктов обучающая выборка состояла из 870 элементов, шести классов  $\{v_1, v_2, v_3, v_4, v_5, v_6\}$  и описывалась шестью признаками, такими как содержание ароматических углеводородов (Aroma), олефинов (Olf), бензолов (ArmBnz), ксилолов (ArmKs), октановое число (RON) и массовая доля МТБЭ (МТВЕ). На рис. 1 представлены лингвистические переменные для трех из рассматриваемых признаков (Olf, Aroma, ArmBnz) и их значения для некоторых классов, полученные при значениях коэффициентов  $\alpha = 0.1$  и  $\beta = 4$ .

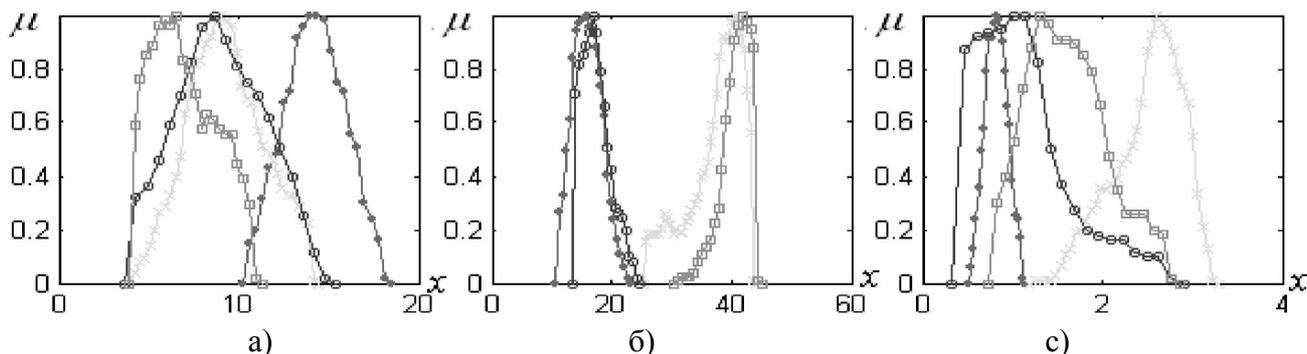


Рис. 1. Значения лингвистических переменных а) Olf, б) Aroma, с) ArBnz для классов образов  $v_1, v_2, v_3, v_4$

На рис. 2 представлен результат работы алгоритма в виде столбчатых диаграмм для образцов  $\bar{x}^{(1)} = (20, 25, 1.3, 0.93, 6)$  и  $\bar{x}^{(2)} = (9, 40, 2.1, 97, 10)$ . Следует отметить, что образец  $\bar{x}^{(1)}$  однозначно является представителем класса образов  $v_1$ , а класс образов, к которому относится  $\bar{x}^{(2)} = (20, 25, 1.3, 0.93, 6)$ , определяется неоднозначно -  $v_3$  или  $v_6$ .

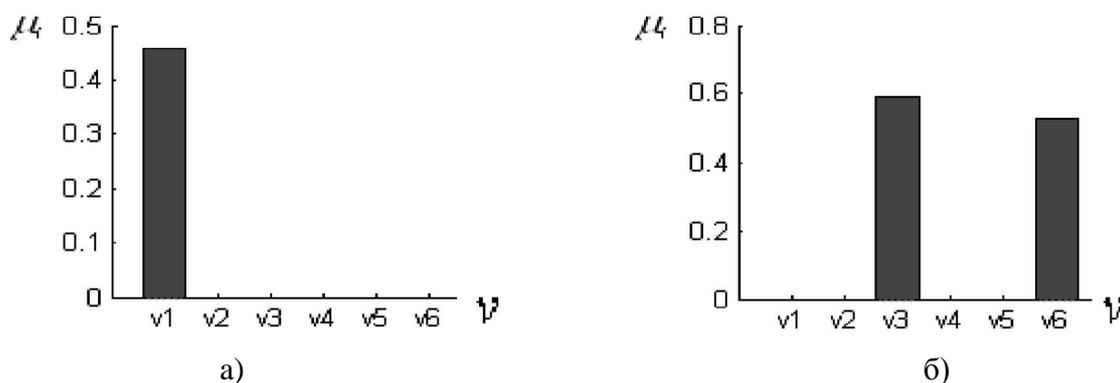


Рис. 2 Результат нечеткого вывода для образцов а)  $\bar{x}^{(1)}$ , б)  $\bar{x}^{(2)}$

#### Адаптация и контроль качества алгоритма

В предлагаемом алгоритме этап адаптации осуществляется за счет выбора параметров  $\alpha$  и  $\beta$  по результатам серии экспериментов. Поиск наилучшего решения затруднен отсутствием общепризнанных универсальных критериев качества решений. На практике для оптимизации небольшого числа параметров используют функционалы скользящего контроля [4]. Фактически методами скользящего контроля измеряется обобщающая способность метода обучения на заданной конечной выборке.

В работе использованы функционал полного скользящего контроля  $Q_c(\mu, X^L) = \frac{1}{N} \sum_{n=1}^N v(\mu(X_n^l), X_n^k)$  и функционал среднего отклонения частоты ошибок на

контроле от частоты ошибок на обучении  $Q_d(\mu, X^L) = \frac{1}{N} \sum_{n=1}^N (v(\mu(X_n^l), X_n^k) - v(\mu(X_n^l), X_n^l))$

алгоритма  $\mu$  по конечной совокупности объектов  $X^L$ , где  $(X_n^l, X_n^k)$ ,  $n=1, \dots, N$  - всевозможные разбиения выборки  $X^L$  на обучающую и контрольную,  $L=l+k$ ;  $v(\mu, X^L)$  - частота ошибок алгоритма  $\mu$  на обучающей выборке  $X^L$ . В результате серии экспериментов для задачи распознавания ЖНП в качестве оптимальных параметров были определены  $\alpha = 0.1$  и  $\beta = 4$ , что обеспечило 92% верно распознанных образцов на тестовой выборке.

### Кластеризация и повышение уровня распознавания

Как уже отмечалось ранее, структура обучающей выборки может быть достаточно сложной. В рассматриваемой задаче в результате визуального анализа диаграмм рассеивания, функций принадлежности нечетких портретов, а также консультаций с экспертом было определено наличие кластеров в некоторых классах образов, в частности для класса  $v_4$ . Это обусловлено технологическими особенностями производства ЖНП. Для класса  $v_4$  были определены два кластера и в базе знаний, соответственно, было определено два правила для данного класса. В результате такой модификации было повышено качество распознавания. На рис. 3 изображена ROC-кривая, для случая, когда в качестве позитивных примеров рассматриваются элементы класса  $v_4$ , а в качестве отрицательных – все остальные примеры. Данная кривая ошибок показывает зависимость количества верно классифицированных положительных примеров от количества неверно классифицированных отрицательных примеров. Качество распознавания полученного алгоритма возросло за счет верно распознанных элементов класса  $v_4$  на 2%.

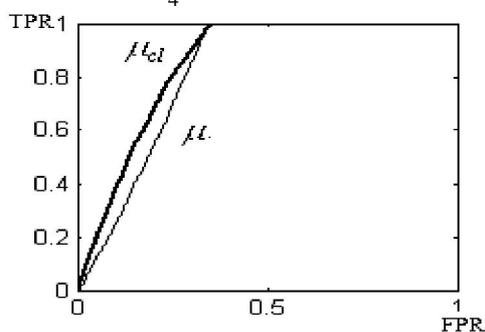


Рис. 3. ROC- кривая для класса  $v_4$

### Выводы

В работе была решена задача классификации ЖНП алгоритмом распознавания на базе нечетких портретов. На данном этапе в лаборатории контроля качества решение принимается профильными специалистами, однако предлагаемый подход позволит автоматизировать процесс и усовершенствовать работу лаборатории. Достоинством предложенного алгоритма является возможность интерпретации результата непосредственно экспертом в ситуациях, когда однозначного ответа не существует. Многие задачи в химической, пищевой промышленности могут быть сведены к описной выше модификации задачи распознавания образов.

В качестве модификации основного алгоритма было выполнено разбиение на кластеры некоторых классов образов, что позволило улучшить качество распознавания на 2% по сравнению с исходным алгоритмом (92% верно распознанных объектов).

### Литература

1. Прикладная статистика: классификация и снижение размерности [Справ. изд.] / С. А. Айвазян, В. М. Бухштабер, И. С. Енюков, А. Д. Мешалкин // Под ред. С.А. Айвазяна. – М.: Финансы и статистика, 1989. – 607 с.: ил.
2. Никифоров И.К. Нейросетевой импедансный метод и устройства идентификации и определения параметров жидких нефтепродуктов: автореф. на соиск. уч. степени канд. тех. наук. – Казань, 2005. – 19 с.
3. Козловский В.А. Решение задачи распознавания по нечетким портретам классов / В.А. Козловский, А.Ю. Максимова // Искусственный интеллект. – 2010. – №4. – С. 221-228.
4. Воронцов К.В. Комбинаторный подход к оценке качества обучаемых алгоритмов. / К.В. Воронцов // Математические вопросы кибернетики. Вып. 13: Сборник статей под ред. О.Б. Лупанова. – М.: Физматлит, 2004. – С.5-36.

УДК 004.93:519.71

## МОДЕЛИ ГРАФИЧЕСКИХ ПРИМИТИВОВ И ИХ КОМПОЗИЦИЙ

**Молчанова В.С.**

Приазовский государственный технический университет, кафедра информатики  
E-mail: vp24@yandex.ru

### **Аннотация**

*Молчанова В.С. Модели графических примитивов и их композиций. Предложен подход к описанию графических примитивов и растровых изображений, формируемых из заданного набора таких примитивов. В его основе лежит синтаксическое описание примитивов и построение автоматов для их генерации или распознавания и графовая модель изображений, допускающая эквивалентные преобразования с целью упрощения модели.*

### **Общая постановка проблемы**

Структурный анализ контуров изображений является одной из основных задач при обработке изображений с целью их интерпретации в системах искусственного интеллекта. В [4] рассмотрен способ представления контура объекта произвольной формы в виде последовательности отрезков прямых, которая в определенном смысле аппроксимирует исходный контур. Такой прием часто используется при решении задач математического анализа и в большом количестве практических приложений.

Однако существует целый класс задач, где представление контуров в виде последовательности отрезков прямых неприемлемо и в лучшем случае приводит к большому количеству дополнительных расчетов, например, при решении задачи распознавания объектов на технических чертежах, схемах печатных плат и т. д. Указанные изображения технического назначения обычно являются комбинациями графических примитивов из достаточно узкого набора, таких, например, как отрезки прямых, дуги окружности или эллипса, возможно, кривые Безье как приближение более сложных линий. При анализе и синтезе таких изображений возникают две взаимосвязанные задачи: описания и распознавания отдельных фрагментов изображения - примитивов, и описания всего изображения в целом как некоторой структуры из таких примитивов.

В работе описывается подход к решению задач описания примитивов и изображения в целом на основе графовых и автоматных моделей. Для упрощения далее рассматриваем такие графические примитивы, которые совпадают со своими контурами.

### **Автоматное описание графических примитивов**

Основная идея автоматного подхода к описанию и распознаванию примитивов опирается на следующие соображения.

Исходное изображение понимается как бинарная прямоугольная матрица размером  $m \times n$ , элементы которой называем ячейками или пикселями, окрашиваемыми на рисунке белым (значение элемента равно 0, фоновый элемент) или черным цветом (значение элемента равно 1, элемент принадлежит собственно изображению). В дальнейшем отождествляем эту матрицу с прямоугольным фрагментом плоскости, разбитым на квадраты (ячейки) со стороной, равной единице, и начало системы координат отнесено к верхней левой точке (соответствует традиционной системе координат, соотнесенной с экраном монитора), а осями абсцисс и ординат будем считать верхнюю и левую границы растра. По этой среде будет перемещаться мобильный агент – автомат, который в процессе

перемещения формирует либо слово описания объекта фрагмента либо решает обратную задачу – формирует изображение по слову описания. Слово описания является заданием на синтез конечного автомата (в виде системы уравнений), который затем способен генерировать нужное слово для восстановления изображения или его фрагмента.

Рассмотрим один из возможных способов перемещения автомата и формирования слова описания. Начиная из начала координат, некоторый автомат-сканер  $A_1$  движется в направлении слева направо сверху вниз, на каждом шаге перемещаясь на соседнюю ячейку. Движение автомата  $A_1$  по РИ продолжается до тех пор, пока ему не встретится точка изображения, окрашенная черным цветом. Координаты  $(x, y)$  этой точки запоминаются и некоторый автомат  $A_2$  начинает движение по внешнему контуру объекта. Под контурной линией объекта (контуром) будем понимать некоторую ломанную, образованную точками растра таким способом, что по крайней мере по одну из сторон каждого отрезка ломанной, расположены пиксели, окрашенные в белый цвет.

В результате движения автомата  $A_2$  по контуру объекта формируется некоторое слово  $P$  в алфавите непроединных символов  $F=\{0, 1, 2, 3, 4, 5, 6, 7\}$  цепочечного кода Фримена [3].

При перемещении автомата  $A_2$  из точки  $(x_i, y_i)$  в точку  $(x_{i+1}, y_{i+1})$  контура к слову  $P$  добавляется один символ, значение которого определяется направлением, в котором было выполнено перемещение (рис. 1).

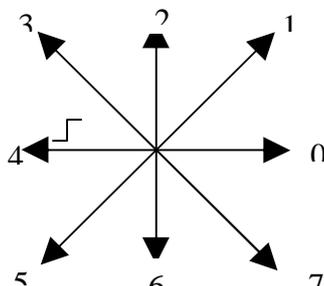


Рис. 1. Непроизводные элементы

Во избежание повторного движения автомата  $A_2$  по уже зафиксированным точкам контура объекта каждая пройденная точка помечается. Автомат  $A_2$  прекращает либо приостанавливает свое движение если:

- 1) Оказался в точке, не имеющей соседей, принадлежащих контуру;
- 2) Оказался в точке из которой начал движение по контуру;
- 3) Оказался в точке, движение из которой, может быть выполнено в нескольких альтернативных направлениях. Такая ситуация представляет особый интерес и будет более подробно рассмотрена ниже.

Под автоматом понимаем автономный инициальный автомат, задаваемый как четверка  $A=(S, Y, \delta, s_0)$ , где  $S, Y$  – множества состояний и выходов соответственно,  $\delta$  – функция переходов,  $s_0$  – начальное состояние [1]. Далее в качестве основного способа задания его функционирования рассматриваем автоматные уравнения с алфавитом  $Y=V$ . Показано [4], что автомат, выполняющий построение отрезка прямой с тангенсом угла наклона  $k$  описывается системой уравнений:

$$\begin{cases} s(t) = [s(t-1) + m] \bmod n, & s(0) = 0 \\ y(t) = \text{sign}(m - s(t-1)), & t > 1 \\ y(1) = 1 \end{cases}, \text{ где } k = m/n \quad (1)$$

Автоматные системы уравнений были предложены для описания дуги эллипса [5]. Основная идея автоматного описания кривых второго порядка опирается на представление состояния автомата как пары  $(S_1, S_2)$  определяющей координаты соответствующей точки контура кривой на растровом изображении; а выходное значение автомата  $y$  – определяет соответствующий элемент из алфавита  $V$ .

$$\begin{cases} s_1(t) = s_1(t-1) + \text{sign}(y(t-1)) \\ s_2(t) = s_2(t-1) + \text{sign}(-y(t-1)) \\ y(t) = \text{case}(t, Y) \\ s_1(0) = 0 \\ s_2(0) = 0 \\ y(0) = 1 \end{cases}, \quad (2)$$

где  $Y = \{-1, 0, 1\}$ ,  $\text{case}(t, Y) = \alpha \Leftrightarrow \min\{f(x, t-1) \mid x \in Y\} = f(\alpha, t-1)$ ,

$f(x, t) = A^2 - (s_1(t) + a)^2 + (B - s_2(t) - b)^2$ , где  $a = 0, b = 1$ , если  $x = -1$ ;  $a = 1, b = 1$ , если  $x = 0$ ;  $a = 1, b = 0$ , если  $x = 1$ .

Описание второй, третьей и четвертой четвертей эллипса базируется на его симметричности относительно оси абсцисс и оси ординат и может быть получено путем проведения некоторых преобразований над каждым элементом слова, описывающего первую четверть.

Автоматные уравнения являются удобным средством генерации слов описания примитивов, по которым затем автомат-плоттер формирует растровое изображение примитивов. Помимо этого, сами слова описания, получаемые либо при движении автомата-сканера по растру, либо порождаемые автоматом-генератором, являются входными словами для автоматов-распознавателей, распознающих тип графического примитива.

### Графовая модель композиции примитивов

Далее исходим из того, что РИ представляет собой композицию базовых примитивов. Точки, в которых происходит соединение двух или более базовых объектов, будем называть характеристическими. Примеры соединения объектов в характеристических точках показаны на рисунке 2.

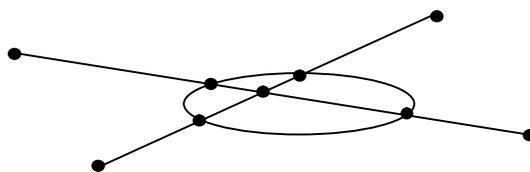


Рис. 2. Примеры соединения базовых объектов в характеристических точках

Выделение характеристических точек осуществляется по правилу: если точка  $C(x, y)$  имеет не менее двух соседних точек с координатами  $C_1(x_1, y_1)$  и  $C_2(x_2, y_2)$ , причем точки  $C_1$  и  $C_2$  не являются соседями (т.е.  $|x_2 - x_1| \geq 2$  или  $|y_2 - y_1| \geq 2$ ), то точка  $C$  характеристическая.

В результате обхода автоматом  $A_2$  объекта на изображении будет получено некоторое конечное множество  $V = \{v_1, v_2, \dots, v_n\}$  характеристических точек и некоторое конечное множество  $S = \{s_1, s_2, \dots, s_m\}$  слов, описывающих фрагменты объекта в алфавите Фримена.

Моделью растрового изображения  $R$  называем пятерку  $M_R = (V, E, \varphi, T, \mu)$ , где тройка  $G = (V, E, \varphi)$  – ориентированный граф [2], в котором множество вершин  $V$  есть множество характеристических точек изображения, функция  $\varphi: E \rightarrow V \times V$  сопоставляет дуге  $e$  пару  $(v_i, v_j)$  характеристических точек, если автомат-сканер перемещался от точки  $v_i$  к точке  $v_j$  по некоторому примитиву  $t_k$ ,  $T$  – множество примитивов (точнее, их имен), а функция  $\mu: E \rightarrow T$

сопоставляет дугам графа отметки - пройденные примитивы, т.е. в данном случае  $\mu(e)=t_k$ . Модель определена с точностью до ориентации дуг. Если ориентация дуги меняется на противоположную, в соответствующей ей примитиве-отметке могут быть изменены по определенному правилу параметры.

Полученной модели сопоставляется некоторое множество путей в нагруженном графе. Путь понимается обычным образом как последовательность вершин графа и отметок соответствующих дуг и может быть определенным образом сокращен с помощью вводимых эквивалентных преобразований. Несколько упрощая, в качестве примера такого преобразования можно привести следующее правило.

Пусть некоторый путь имеет подпуть вида  $v_i T_k v_j T_m v_l$ , где  $v_i, v_j, v_l$  – вершины, а  $T_k, T_m$  примитивы. Если  $T_k = T_m = T$ , то этот подпуть может быть заменен на подпуть  $v_i T v_l$ . Пример исходного и минимизированного графа для РИ, приведенного на рисунке 2, без учета нагруженности показан на рисунке 3 (а,б).

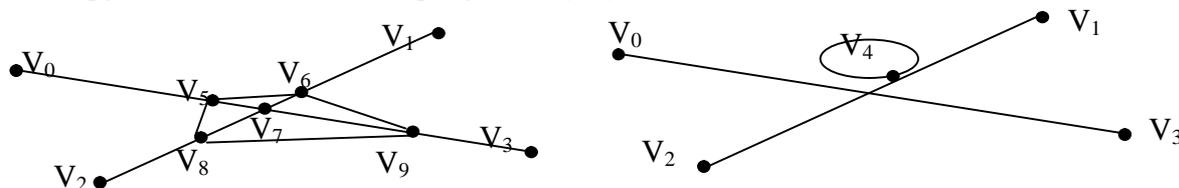


Рис. 3. А) Графовая модель растрового изображения, показанного на рис. 2 до минимизации. Б) Графовая модель растрового изображения, показанного на рис. 2 после минимизации.

Дополнительное упорядочение дуг графа по типам примитивов и их параметрам позволяет упростить такие преобразования и уменьшить совокупную длину указанных путей и, таким образом, упростить и первичную модель изображения. Эквивалентность преобразований гарантирует однозначность восстановления исходного изображения по преобразованной модели.

### Выводы

Предложенная модель позволяет получить компактное векторное представление некоторых классов растровых изображений и, в совокупности с автоматным представлением примитивов, минимизировать вычислительные затраты при синтезе образов. Кроме того, заметим, что структура графовой модели не меняется при аффинных преобразованиях, а ее параметры при этом легко модифицируются.

### Литература

1. Кудрявцев В. Б., Алешин С. В., Подколзин А. С. Введение в теорию автоматов – М.: Наука. Гл. ред. физ. – мат. лит., 1985. – 320 с.
2. Н. Кристофидес. Теория графов. Алгоритмический подход – М.: Мир, 1978. – 427 с.
3. Фурман Я.А.. Введение в контурный анализ; приложения к обработке изображений и сигналов/ Я.А.Фурман, А.В Кривецкий, А.К. Передреев, А.А. Роженцов, Р.Г. Хафизов, И.Л Егошина, А.Н. Леухин; Под ред. Я.А. Фурмана. — 2-е изд., испр. — М.: ФИЗМАТЛИТ, 2003. — 592 с.
4. Ю.Б.Деглина, В.С. Денисова, В.А.Козловский. Автоматные алгоритмы синтеза образов// Искусственный интеллект, 2008 - №3 – с.290-295
5. Козловский В.А. Построение автоматных моделей простейших графических примитивов / В.А. Козловский, В.С. Молчанова// Искусственный интеллект, 2010- №4. – С. 375-380.

УДК 517.9, 004.032

**РАЗРАБОТКА СИСТЕМЫ ПРИНЯТИЯ РЕШЕНИЙ  
В ПРОЦЕССАХ БИОСОРБЦИИ СЕРЕБРА****Новаковская А.О.**

Донецкий национальный технический университет, кафедра прикладной математики и информатики

E-mail: [my\\_anna@mail.ru](mailto:my_anna@mail.ru)**Аннотация**

*Новаковская А.О. Разработка системы принятия решений в процессах биосорбции серебра. Разработка системы принятия решений в процессах биосорбции серебра. Рассмотрены принципы принятия решений в трехуровневой системе управления участком извлечения серебра из вторичных растворов. Представлены принципы формирования управляемой нейросети, на основе которой осуществляется принятие решений по эффективному управлению процессами.*

**Введение**

Традиционно, серебро из вторичных видов сырья извлекается с помощью специально приготовленных сорбентов – смол, стоимость которых составляет 2000-2500 у.е./тонну. В современных условиях применение смол обуславливает нерентабельность вторичного производства серебра. Альтернативой этому типу сорбентов являются плесневые отходы производства антибиотиков, которые получили название «биосорбенты». Целью данного исследования является разработка системы принятия решений в процессах биосорбции серебра. Разработка включает в себя создание математических моделей, постановку и формализацию задач управления, создание и численное исследование алгоритмов системы управления.

Физико-химические особенности извлечения серебра с помощью биосорбентов исследованы в работе [1]. Математические модели процессов биосорбции, восстановления и десорбции представлены в работах [2-4].

**Основная часть**

Разрабатываемая система принятия решений предназначена для управления участком извлечения серебра из вторичных растворов, содержащих около 200 мг/л серебра. Технология организована следующим образом. Раствор последовательно перетекает через 9 колонн, заполненных гранулами биосорбента, установленных с понижением уровня. В первой группе находятся три колонны, в трех последующих – по две колонны.

Подсистема принятия решений предусматривает реализацию управления на трех уровнях. На первом уровне осуществляется прогноз и управление кинетикой процессов, протекающих в каждой колонне. Исходя из физико-химических особенностей процесса, выделены режимы процесса биосорбции S, длящегося около 2 часов, после чего подача раствора через колонну прекращается и колонна переходит в режим восстановления технологических свойств сорбента (R), длящегося около 8 часов. В соответствии с этим выделенные режимы S-R составляют фазу функционирования колонны. В зависимости от типа плесневой культуры, различные типы сорбентов могут выдерживать 5-8 технологических фаз. Предусматривается чередование режимов и фаз в колоннах каждого уровня. При этом, колонны первых трех уровней извлекают основную массу серебра, колонны последних уровней предназначены для доочистки раствора от серебра, подчиненных санитарным нормам сливных растворов. Одновременно функционируют три последовательно соединенные колонны, каждая из которых работает в соответствующем

режиме (S или R). Разработана матрица смежности, определяющая порядок последовательности соединения колонн. При этом первые колонны в одном технологическом цикле работают 10 часов. Для каждого из указанных режимов разработаны и идентифицированы динамические модели прогноза текущего содержания серебра в растворе, расходования биосорбента и извлечения серебра из раствора. Конечные условия решения уравнений модели прогноза содержания серебра в растворе, проходящем через колонну, являются начальными условиями задачи Коши для последующей колонны. Таким образом, динамические модели режимов позволяют итерационно оценивать состояние объекта управления – колонны, как в режиме сорбции, так и в режиме восстановления. Согласно современной методологии, параметрические дифференциальные уравнения математической модели, позволяющей оценивать состояния объекта управления, могут быть представлены в виде слоя нейросети, имеющего несколько входов и выходов. Таким образом, девять технологических сорбционных колонн представлены в виде девятислойной нейросети, обучение которой осуществляется классическими методами идентификации параметров. Завершение режимов определяется по заданной точности итеративной оценки состояния биосорбента, характеризующей переход плесневой культуры в режим насыщения.

Завершение режима восстановления технологических свойств сорбентов подчинено логическому правилу (1)

$$\forall A_{ij} \forall k \exists G_{i+1,j}^{Ag}(k) : \left( G_{i+1,j}^{Ag}(k) = D_{i+1,j} \right) \vee \left( G_{i,j+1}^{Ag}(k) = D_{i,j+1}(k) \right), \quad (1)$$

где  $A_{ij}$  – колонна,  $i$  – номер уровня,  $j$  – номер колонны,  $k$  – шаг итерации,  $G_i^{Ag}$  – количество серебра, усвоенного сорбентом колонны  $A_{ij}$ ,  $D_{ij}$  – динамическая обменная емкость биосорбента, прогнозируемая по математической модели.

После того, как данная плесневая культура, используемая в качестве биосорбента, отработала фиксированное для каждой культуры число технологических циклов, осуществляется перевод колонны на режим десорбции согласно логическому правилу (2):

$$\forall A_{ij} \forall k \exists G_{ij}^{Ag}(k) : G_{ij}^{Ag}(k) \leq 0,5G_{cp}^{Ag}, A_{ij} \notin k, \quad (2)$$

где  $G_{гр}$  – предельное значение серебра, которое способен усвоить сорбент.

В результате чистое серебро извлекается из биосорбента и в данной колонне производят замену биосорбента, помещенного в сетку, на новую порцию.

Условия переключения колонн на различные режимы, фазы, циклы, осуществляется на втором уровне управления в подчинении приведенным логическим правилам и в соответствии с матрицей смежности, то есть управление осуществляется путем переключения нейронов между слоями нейросети.

Таким образом, на втором уровне осуществляется дискретное управление режимами переключения нейронов и слоев нейросети, то есть в системе применяется управляемая нейросеть. Динамический прогноз на втором уровне осуществляется с помощью математической модели, представленной системой параметрических дифференциальных уравнений следующего вида.

Уравнение прогноза скорости усвоения серебра сорбентом:

$$\frac{dG^{Ag}(A_{ij})}{dt} = \sum_{l=1}^k G_l^{Ag}(A_{ij}, T_{1,l}), \quad (3)$$

$$G_l^{Ag} (A_{ij}, T_{1,l}) = G_l^{RCOOAg} (A_{ij}, T_{1,l}) + G_l^{RSAg} (A_{ij}, T_{1,l}) + G_l^{RAgNO_3} (A_{ij}, T_{1,l}), \quad (4)$$

где  $l$  – номер цикла,  $l = \overline{1, k}$ ,  $T_{1,l}$  – длительность режима сорбции,  $G^{Ag}$  – количество серебра, усвоенного сорбентом за все циклы,  $G_l^{Ag}$  – количество серебра, усвоенного за один цикл,  $G_l^{RCOOAg}$ ,  $G_l^{RSAg}$ ,  $G_l^{RAgNO_3}$  – количество серебра, усвоенного активными центрами сорбента RCOOH, RSH, R за один цикл, соответственно.

Уравнения скорости роста активных центров сорбента:

$$\frac{dG^{At} (A_{ij})}{dt} = \sum_{l=1}^k G_l^{At} (A_{ij}, T_{2,l}), \quad (5)$$

где  $G^{At}$  – общее количество активных центров сорбента,  $T_{2,l}$  – длительность режима восстановления.

$$G_l^{At} (A_{ij}, T_{2,l}) = G_l^{RCOOH} (A_{ij}, T_{2,l}) + G_l^{RSH} (A_{ij}, T_{2,l}) + G_l^R (A_{ij}, T_{2,l}), \quad (6)$$

где  $G_l^{RCOOH}$ ,  $G_l^{RSH}$ ,  $G_l^R$  – количество активных центров сорбента RCOOH, RSH, R, восстановленных за один цикл, соответственно.

Назначение системы второго уровня – максимально использовать технологические свойства плесневой культуры до осуществления ее замены, что позволяет сокращать технологическую себестоимость производства.

На третьем уровне системы принятия решений рассматривается экономическая целесообразность закупки того или иного вида отходов производства антибиотиков, каждый из которых характеризуется своей ценой, сорбционной способностью, количеством технологических циклов использования этой культуры в качестве сорбента. Для прогноза принимаемых решений разработана статическая статистическая модель верхнего уровня управления.

Уравнение прогноза использования сорбентов различных типов:

$$y_1 = -1,97 + 0,39x_{11} + 0,22x_{12} + 0,11x_{13} + 0,13x_{14} + 0,08x_{15} + 0,07x_{16}, \quad (7)$$

где  $x_{ij}$  – тип сорбента,  $i$  – номер переменной,  $i = \overline{1, 8}$ ,  $j$  – номер типа сорбента,  $j = \overline{1, 6}$ .

Уравнение прогноза стоимости закупки сорбента:

$$y_2 = 22,5 + 1,4x_{21}x_{31} + 1,2x_{22}x_{32} + 0,92x_{23}x_{33} + 0,84x_{24}x_{34} + 0,65x_{25}x_{35} + 0,47x_{26}x_{36}, \quad (8)$$

где  $x_{2j}$  – вес сорбента,  $x_{3j}$  – цена сорбента.

Уравнение прогноза предельных возможностей использования сорбента:

$$\begin{aligned} y_3 = & -153,8 + 0,51x_{41} + 0,42x_{42} + 0,18x_{43} + 0,21x_{44} + 0,16x_{45} + \\ & + 0,09x_{46} + 1,6x_{51} + 1,7x_{52} + 1,2x_{53} + 0,9x_{54} + 0,7x_{55} + 0,6x_{56} + \\ & + 0,32x_{41}x_{51} + 0,34x_{42}x_{52} + 0,24x_{43}x_{53} + 0,23x_{44}x_{54} + \\ & + 0,14x_{45}x_{55} + 0,11x_{46}x_{56} \end{aligned} \quad (9)$$

где  $x_{4j}$  – предельная допустимая обменная емкость сорбента (ПДОЕ),  $x_{5j}$  – количество циклов использования сорбента.

Уравнение прогноза кинетики  $j$ -го сорбента:

$$\begin{aligned} y_4 = & -209,6 + 0,63x_{41} + 0,61x_{42} + 0,57x_{43} + 0,51x_{44} + 0,42x_{45} + 0,35x_{46} + \\ & + 3,1x_{61} + 2,7x_{62} + 1,3x_{63} + 1,2x_{64} + 1,1x_{65} + 1,2x_{66} + 0,8x_{41}x_{61} + 0,9x_{42}x_{62} + \\ & + 0,6x_{43}x_{63} + 0,5x_{44}x_{64} + 0,4x_{45}x_{65} + 0,3x_{46}x_{66} \end{aligned} \quad (10)$$

где  $x_{6j}$  – время насыщения  $j$ -го сорбента.

Уравнение прогноза количества серебра, извлекаемого сорбентом:

$$y_5 = -160,64 + 0,71x_{41} + 0,69x_{42} + 0,65x_{43} + 0,57x_{44} + 0,43x_{45} + 0,31x_{46} + \\ + 0,081x_{71} + 0,079x_{72} + 0,071x_{73} + 0,064x_{74} + 0,061x_{75} + 0,057x_{76} + \\ + 0,0061x_{41}x_{71} + 0,0048x_{42}x_{72} + 0,0043x_{43}x_{73} + 0,0039x_{44}x_{74} + \\ + 0,0028x_{45}x_{75} + 0,0023x_{46}x_{76} \quad (11)$$

где  $x_{7j}$  – количество извлекаемого серебра из  $j$ -го сорбента.

Уравнение прогноза длительности использования сорбента:

$$y_6 = 65,6 + 0,49x_{41} + 0,52x_{42} + 0,38x_{43} + 0,36x_{44} + 0,29x_{45} + 0,21x_{46} + \\ + 0,041x_{81} + 0,049x_{82} + 0,032x_{83} + 0,022x_{84} + 0,021x_{85} + 0,015x_{86} + \\ + 0,0031x_{41}x_{81} + 0,0035x_{42}x_{82} + 0,0024x_{43}x_{83} + 0,0027x_{44}x_{84} + \\ + 0,0021x_{45}x_{85} + 0,0017x_{46}x_{86} \quad (12)$$

где  $x_{8j}$  – длительность использования сорбента.

Уравнение прогноза технико-экономических показателей сорбента:

$$y_7 = 63,8 + 1,23y_1 + 0,051y_2 + 0,029y_3 + 0,023y_4 + 0,031y_5 \quad (13)$$

Приведенные уравнения второго и третьего уровней позволяют прогнозировать состояние технологических агрегатов участка биосорбции и принимать решения по управлению режимами (второй уровень) и закупке сорбента (третий уровень).

Уравнение (13) используется в качестве функционала цели, оптимальные значения которого позволяют принимать решения о количестве и цене закупки сорбентов различных типов.

## Выводы

В научном исследовании создана математическая модель и алгоритм трехуровневой системы принятия решений, практическая значимость которых позволяет использовать отходы производства антибиотиков стоимостью около 30 у.е./тонна, что делает рентабельным процесс извлечения серебра из вторичных растворов. Научная новизна работы представлена разработанными динамическими моделями первого и второго уровней, статического моделью третьего уровня, управляемой девятислойной нейросетью оценки состояний объекта управления и циклического переключения технологических агрегатов.

## Литература

1. Стрижко Л.С. Металлургия благородных металлов / Л.С. Стрижко. - М.: МИСиС, 2001.- 336с.
2. Криводубский О.А. Прогнозирование процессов биосорбции серебра / О.А. Криводубский, А.О. Новаковская // Системи обробки інформації: сб. наук. пр. Харківського університету Повітряних Сил ім. Івана Кожедуба. – Вып. 2 (76). – Х., 2009. – С. 113 – 117.
3. Криводубский О.А. Математическая модель восстановления активных свойств сорбента / О.А. Криводубский, А.О. Новаковская // Інформатика, кібернетика та обчислювальна техніка: сб. наук. пр. Донецького нац. технічного ун-ту. – Вып. 10 (153). – Д., 2009. – С. 251 – 254.
4. Криводубский О.А. Прогнозирование процесса десорбции серебра / О.А. Криводубский, А.О. Новаковская // Радіоелектронні і комп'ютерні системи: сб. науч. тр. Нац. аэрокосм. ун-та им. Н.Е. Жуковского «ХАИ». - Вып. 1 (35). – Х., 2009. – С. 93 – 99.

УДК 004.9

**ОБНАРУЖЕНИЕ ВДАВЛЕННЫХ НОМЕРОВ НА ОСНОВЕ ТЕКСТУРНОЙ СЕГМЕНТАЦИИ****Тимощенко П.А.**

Днепропетровский национальный университет имени Олеся Гончара  
кафедра автоматизированных систем обработки информации  
E-mail: t\_pa@mail.ru

**Аннотация**

**Тимощенко П.А. Обнаружение вдавленных номеров на основе текстурной сегментации.** В работе рассматривается обнаружение области номера колеса на статическом кадре, используя метод текстурного анализа на основе распределений локальных бинарных паттернов.

**Общая постановка проблемы**

В процессе контроля качества железнодорожных колес необходимо вносить в базу данных номер контролируемого колеса. На сегодняшний день процесс считывания номеров производится вручную, что замедляет технологический процесс, приводит к возможным ошибкам в данных. Номер состоит из групп цифр и символов от 1 до 5 в группе, расположенных на ободе колеса [ГОСТ 10791].

Основными проблемами, которые стоят на пути автоматизации процесса считывания являются:

- 1) сложность обнаружения номера (номер нанесен на поверхность обода в горячем состоянии и имеет тот же цвет, что и обод)
- 2) глубина символа до 4 мм и может изменяться в пределах группы символов
- 3) грани символа сглажены различным образом (в зависимости от способа нанесения номера)
- 4) перегородка, отделяющая два символа в группе, может иметь изменяющуюся высоту.

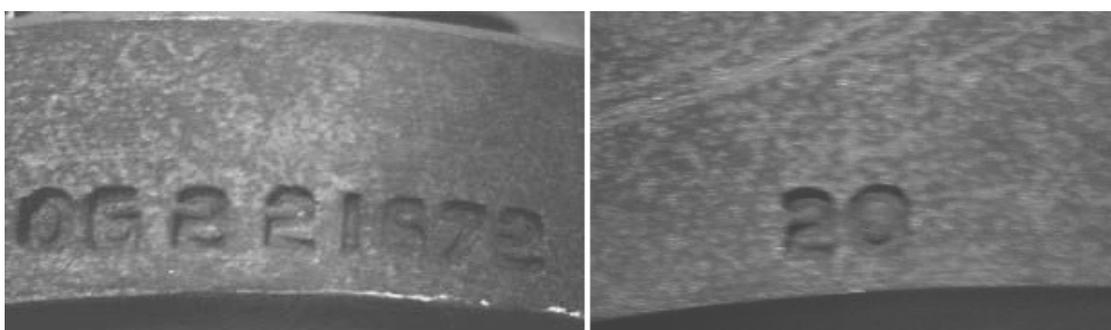


Рис. 1. Пример изображения обода железнодорожного колеса с частью номера

Данная работа посвящена решению первой проблемы, а именно выделению области поиска символов на ободе. Для этого предлагается рассматривать поверхность колеса, как одну текстуру, а область номера – вторую. Таким образом, задача сводится к сегментации изображения на два класса.

Сегментацию предлагается проводить на основе анализа распределений локальных бинарных паттернов (LBP)[1]. LBP – это структурный дескриптор, позволяющий провести перекодировку окрестности некоторой точки. Пример вычисления паттерна для окрестности

3x3 изображен на рис.2[2]. Каждой точке в окрестности ставится в соответствие 1, если ее значение больше значения центральной точки, иначе ставится 0 (рис.2б). Результат умножается на число, равное 2 в степени номер точки в окрестности (рис. 2в). Затем вычисляется сумма, которая и будет соответствовать паттерну.

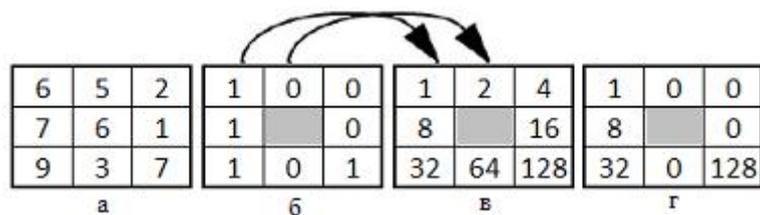


Рис.2. Пример вычисления LBP. (а)- исходные значения в окрестности; (б) – результат бинаризации; (в) – степени числа 2, соответствующие каждой позиции; (г) – результат наложения б на в.  $LBP = 1 + 8 + 32 + 128 = 169$ ,  $C = (6 + 7 + 9 + 7) / 4 - (2 + 1 + 3 + 5) / 4 = 4.5$ ,  $M = (6 + 5 + 2 + 7 + 6 + 1 + 9 + 3 + 7) / 9 = 5.1$

В качестве дополнения паттерна вычисляется величина контраста  $C$  или среднего  $M$ . Затем строится двумерная гистограмма значений пар  $LBP/C$  или  $LBP/M$ . В данной работе используется  $LBP/C$ . Для контраста, как правило, выбирают число интервалов равное 8. Эксперименты показали, что при числе интервалов 4 и 16 результат сегментации получается хуже, чем при 8. Для  $LBP$  число ячеек выбирается в соответствии с возможным количеством значений паттерна, для окрестности 3x3 максимальное значение равно 256. Таким образом, гистограмма локальных бинарных паттернов имеет размер 256x8.

Для сравнения двух гистограмм используются различные критерии, но наиболее популярным является логарифмическое отношение правдоподобия ( $G$ -статистика). Значение  $G$  показывает вероятность того, что два распределения принадлежат одному классу: чем выше значение  $G$ -статистики, тем меньше вероятность принадлежности двух распределений к одному классу. Значение  $G$  вычисляется по формуле:

$$G = 2 \left[ \sum_{s,m} \sum_{i=1}^n f_i \log f_i \right] - \left[ \sum_{s,m} \left( \sum_{i=1}^n f_i \right) \log \left( \sum_{i=1}^n f_i \right) \right] - \left[ \sum_{i=1}^n \left( \sum_{s,m} f_i \right) \log \left( \sum_{s,m} f_i \right) \right] + \left[ \sum_{s,m} \left( \sum_{i=1}^n f_i \right) \log \left( \sum_{s,m} \left( \sum_{i=1}^n f_i \right) \right) \right] \quad (1)$$

### Общий алгоритм сегментации

Общепринятый алгоритм сегментации состоит из трех основных этапов[2,3]:

- 1) иерархическое разбиение изображения
- 2) объединение похожих областей
- 3) корректировка границ

Однако в силу ряда особенностей задачи, ключевая из которых – неоднородность освещения, предлагается предварительно разбить по вертикали изображение на несколько полос, с небольшим пересечением. После проведения объединения областей в каждой полосе проводить объединение соответствующих областей из разных полос. Иллюстрация этапов работы алгоритма показана на рис 3.

### Иерархическое разбиение

Изображение разбивается на полосы шириной 70 пикселей – пятая часть ширины и примерная ширина двух подряд идущих символов. На выбор ширины полосы также влияют

параметры фильтрации и распределение освещения. Эксперименты показали, что допустимые значения лежат в диапазоне от 50 до 100, при выборе которых результат сегментации существенно не меняется.

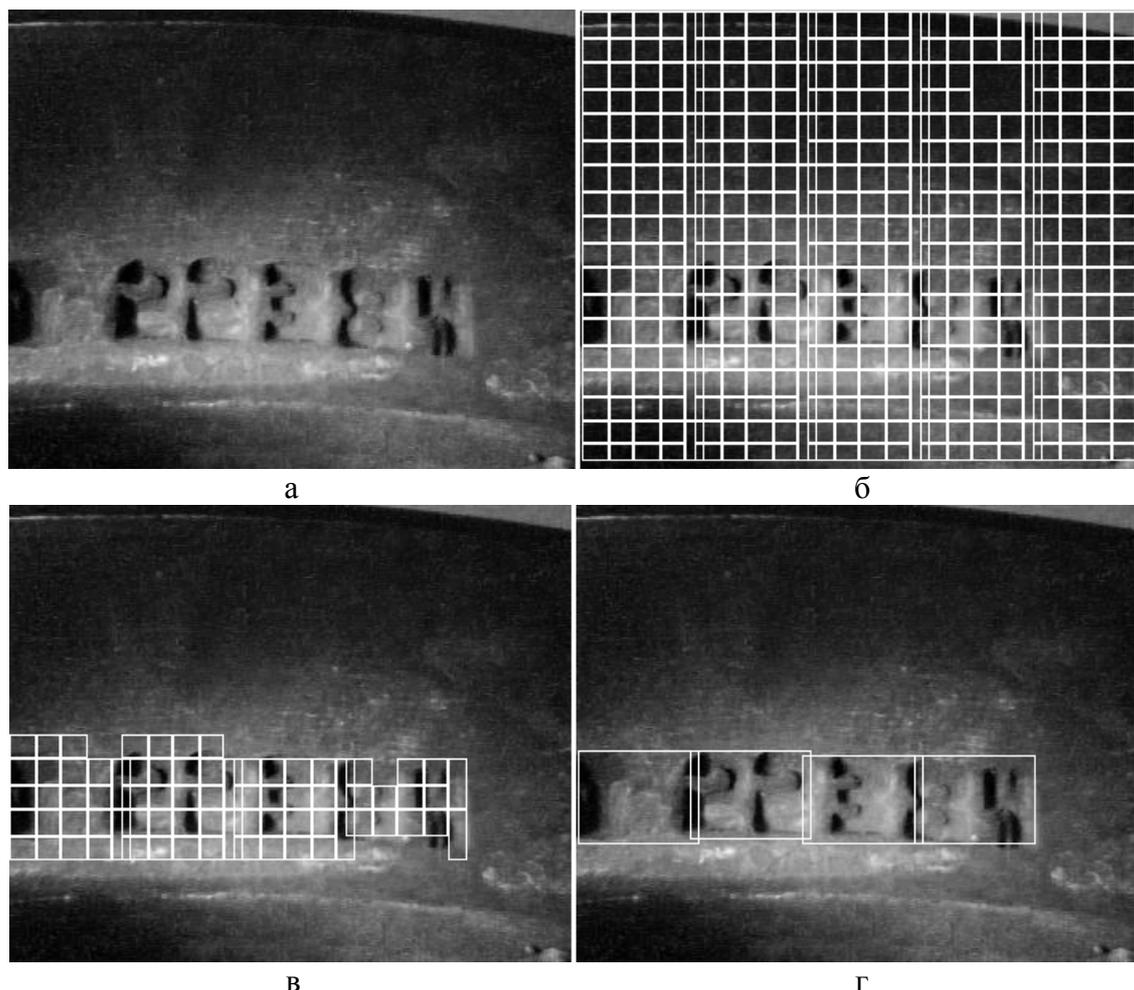


Рис.3. Этапы работы алгоритма сегментации: (а) – исходное изображение; (б) – иерархическое разбиение всех полос с пересечением равным двум размерам окна для расчета LBP; (в) – выделение нужной области; (г) – результат корректировки границ и представление каждой области, как прямоугольника.

Каждая полоса разбивается на области размером  $S_{max}$ , которые проверяются на однородность по следующему алгоритму. Область разбивается на 4 квадранта, и для каждой пары квадрантов вычисляется значение  $G$  (6 значений). Область считаем не однородной, если выполняется следующее:

$$R = \frac{G_{max}}{G_{min}} > X \quad (2)$$

Величина  $X$  характеризует степень однородности и выбирается в пределах 1,1-1,2. Если область неоднородна, то каждый квадрант считаем отдельной областью и для него проводим соответствующие вычисления. Если размер квадранта области достигнет пороговых размеров, то дальнейшее разбиение не имеет смысла и область считается однородной. В работе максимальный размер области выбран равным 64 пикселя, а минимальный равным 16. Результат разбиения показан на рис.3б.

### Объединение областей

Для всех пар областей, имеющих общую границу, подсчитывается мера важности (MI), которая определяется выражением:

$$MI = p \cdot G \quad (3)$$

где  $p$  – площадь меньшей из областей;  $G$  – критерий подобия по формуле (1).

На каждой итерации выделяется пара областей, для которых MI минимально. Найденные области объединяются, и пересчитывается MI для всех областей, граничащих с вновь образованной областью.

Существующий критерий остановки[2], когда соотношение текущего значения и максимального превышает какой-то порог, для данной задачи не подходит, так как область может содержать только один класс. Кроме того, этот порог может существенно изменяться для каждой полосы. Поэтому в качестве первого критерия остановки предлагается разбиение на 3 класса. Априори считается, что область, которая лежит на границе с краем колеса, не содержит символов. Таким образом, выделяется 15 областей, 5 из которых заведомо не содержат символов и условно объединяются в одну область. Явного объединения делать не нужно, так как корректировка границ будет производиться в пределах каждой полосы. Для определения области интереса применяется алгоритм на основе подсчета MI. Результат обнаружения области интереса представлен на рис. 3в.

### Корректировка областей

Область интереса очерчивается прямоугольником в пределах полосы. Верхняя и нижняя границы вычисляются как среднее значение по соответствующей границе. Две строки, которые лежат на границе (внутри и снаружи области) проверяются на принадлежность двум областям по 16 строк. Если после проверки граница изменилась, то процесс повторяется. Результат корректировки границ представлен на рис.3г. Объединение выделенных областей является результатом сегментации, который будет использоваться для дальнейшей обработки.

### Выводы

Использование задачи сегментации текстур позволило выделить область номера на снимке ступицы железнодорожного колеса, что является важной задачей на этапе настройки системы считывания номеров в режиме реального времени. Представленный алгоритм позволяет решать задачу выделения символов простыми расчетами и выбором трех критериев, небольшое изменение которых существенно не влияет на результат, что позволяет указать их явно, не создавая дополнительных алгоритмов адаптации.

### Литература

5. T. Ojala, M. Pietikainen and D. Harwood, "A comparative study of texture measures with classification based on feature distributions," *Pattern Recognition* **29** (1996) 51-59.

6. T. Ojala and M. Pietikainen, "Unsupervised texture segmentation using feature distributions," *Pattern Recognition* **32** (1999) 477-486.

7. The Computer Science and Engineering CSE Laboratory/ Интернет-ресурс. - Режим доступа : <http://www.cse.oulu.fi/MVG/Research/LBP>

УДК 004.382.7(075)

**АРХИТЕКТУРЫ И ТЕНДЕНЦИИ РАЗВИТИЯ СЕТЕВЫХ ПРОЦЕССОРОВ****Моатаз Юнис, Ладыженский Ю.В.**

Донецкий национальный технический университет

Кафедра прикладной математики и информатики

E-mail: [LY@cs.dgtu.donetsk.ua](mailto:LY@cs.dgtu.donetsk.ua)**Аннотация**

*Моатаз Юнис, Ладыженский Ю.В. Архитектуры и тенденции развития сетевых процессоров. Проведен анализ архитектур и тенденций развития сетевых процессоров. Сформулированы некоторые актуальные нерешенные проблемы проектирования и исследования сетевых процессоров.*

**Введение**

В компьютерных сетях используется широкий спектр аппаратно-программных систем, предназначенных для обработки пакетов данных. К ним относятся коммутаторы, маршрутизаторы, процессоры трансляции сетевых адресов, системы обнаружения вторжений, брандмауэры, ADSL-модемы [1]. Сетевые системы проектируются по критериям высокой производительности и широких функциональных возможностей в условиях ограничений на стоимость, физические размеры и время изготовления с целью скорейшего выхода на рынок [2].

Разрабатываемые системы должны быть масштабируемыми, достаточно универсальными и гибкими. В ходе проектирования необходимо учитывать быстрые изменения в рыночных тенденциях, применяемых технологиях и технических требованиях к выпускаемой системе.

**Стадии развития компьютерных сетей**

С момента появления компьютерных сетей 40 лет назад они прошли несколько стадий развития. Основные сетевые устройства первого поколения (с 1975- 1980г.г.) были универсальными компьютерами. Маршрутизатор был мини компьютером с программой, реализующей IP протокол. Во втором поколении (с 1990-95г.г.) появились специализированные процессоры для классификации пакетов и некоторых других функций, стали применяться высокоскоростные коммутаторы. Но большинство функций сетевой обработки выполнялось на универсальных процессорах. Третье поколение сетевых устройств (с 2000г.) обладает полностью распределенной архитектурой, реализуемой на проблемно ориентированных СБИС, содержащими универсальный процессор, выделенные процессоры для каждого сетевого интерфейса. Это поколение отличается существенным ускорением обработки данных [3].

Переход в третьем поколении к полностью распределенной архитектуре усложнил разработку и исследование свойств сетевых систем. Необходимость поддерживать в мультипроцессорных маршрутизаторах отдельные маршрутные таблицы для каждого процессора на каждом сетевом интерфейсе создает дополнительные проблемы синхронизации репликаций таблиц при обновлении. Кроме того, маршрутизатор должен совмещать обновление маршрутных таблиц с обработкой пакетов данных.

В архитектурах третьего поколения возникли новые проблемы: высокая стоимость устройств на СБИС с проблемно-ориентированными процессорами, длительное время (18 - 24 месяца) выпуска новых изделий на рынок, сложность тестирования и верификации пакетных процессоров, недостаточная гибкость применяемых СБИС.

Негибкость СБИС приводит к существенным дополнительным затратам времени на значительное перепроектирование микросхемы даже в случае малых изменений в требованиях к проекту. Кроме того, модификация и настройка СБИС для использования в новых сетевых системах увеличивает стоимость и время разработок.

### **Программируемые сетевые процессоры**

Бурное развитие компьютерных сетей конце 1990-х годов привело к быстрым изменениям к сетевым системам и применению нового подхода для их создания: разработка программируемых процессоров, ориентированных на задачи обработки сетевых пакетов.

Идея архитектуры состоит в комбинации программируемости сетевых систем первого поколения с высокой скоростью систем третьего поколения. Устройства с указанной архитектурой называются сетевыми процессорами. Чтобы сетевые процессоры обладали достаточной производительностью, задачи обработки сетевых пакетов должны быть детально проанализированы, выполнена их функциональная декомпозиция, сделаны оценки их временной и емкостной сложности [4].

Для наиболее времяемких задач в структуру сетевых процессоров нужно включать проблемно ориентированные аппаратные решения. Использование сетевых процессоров оказывается экономически выгодным в системах, содержащих много сетевых интерфейсов, потоки пакетов через которые обрабатываются мультипроцессорными устройствами.

### **Базовые архитектуры сетевых процессоров**

Для достижения высокой производительности в современных сетевых процессорах применяются следующие базовые архитектурные решения: однопоточный процессор с высокой тактовой частотой, параллеливание потоков пакетов, конвейерная обработка пакетов.

Возможности роста производительности первой архитектуры в настоящее время практически исчерпаны.

Особенностью параллельной архитектуры является необходимость выбора и реализации аппаратного механизма эффективного управления входной очередью пакетов[2].

В конвейерной архитектуре поток пакетов перемещается по линии функциональных блоков, каждый из которых выполняет свою часть требуемой обработки пакета.

В современных сетевых процессорах применяются и комбинированные конвейерно-параллельные или параллельно-конвейерные архитектуры.

Внутри сетевого процессора выделяются два основных потока данных:

- пакеты, требующие простой обработки с высокой скоростью
- пакеты, требующие сложной обработки на меньшей скорости.

Для каждого потока может быть выделена своя аппаратная подсистема сетевого процессора.

В докладе анализируются достоинства и недостатки архитектур процессоров, разработанных компаниями Intel, Cisco, Alchemy, Agere, IBM.

### **Применение сетевых процессоров в сетевых взаимодействиях**

Можно выделить три уровня сетевых взаимодействий, на которых можно использовать сетевые процессоры:

- базовый уровень для передачи больших объемов данных на высокой скорости. Здесь реализуются функции маршрутизации, коммутации, контроля доступа.
- граничный межсетевой уровень. На границах происходит вход на базовый уровень и выход из него. Функции граничного уровня отличаются высокой сложностью, выполняются со средней и высокой скоростью, включают маршрутизацию, коммутацию, управление потоком, контроль доступа, управление качеством обслуживания [2].
- уровень доступа. Этот уровень включает все точки доставки информации в сети. Конечные пользователи получают доступ в Интернет сети через локальные сети, широкополосные (данные, видео, звук) сети, телефонные каналы. На этом уровне существует много относительно медленных различных протоколов и технологий взаимодействия.

### **Основные функции сетевых процессоров и требования к ним**

Анализ протоколов и выпускаемых сетевых систем показывает, что основными функциями сетевых процессоров являются:

- анализ заголовков пакетов, сравнение полей заголовков с шаблонами, поиск и выборка из таблиц
- перенаправление входящих пакетов на выходные порты
- контроль доступа и управление очередями пакетов
- ограничение выходного потока пакетов и управление исходящим трафиком
- модификация передаваемых пакетов.
- Основными требованиями к сетевым процессорам являются:
- высокая скорость обработки данных, близкая к скорости передачи данных по сети
- гибкость и программируемость
- быстрый цикл разработки для выпуска на рынок
- удобство обслуживания конечных пользователей в точках доставки информации в сеть

### **Тенденции развития и проблемы проектирования сетевых процессоров**

Наиболее важными тенденциями в области теории и реализации сетевых процессоров являются следующие:

- реализация сетевых процессоров с реконфигурируемой архитектурой на FPGA
  - создание параметризуемых аппаратных платформ для архитектур с множеством сетевых процессоров
  - разделение потоков пакетов путем фильтрации и оптимизация путей обработки разделенных потоков
  - разработка операционных систем для сетевых процессоров
  - применение асинхронных архитектур для ускорения обработки пакетов
  - создание сетевых процессоров для мобильных сетей и сетей специального назначения
- Основными проблемами проектирования сетевых процессоров являются:
- определение наиболее важных задач обработки пакетов для заданных сетевых протоколов
  - определение критериев оптимизации архитектур процессоров

- определение состава, структуры и способов взаимодействия функциональных блоков, обеспечивающих повышение быстродействия
- выбор эффективных интерфейсов ввода-вывода
- определение оптимальных технологий и объемов памяти программ и памяти данных
- выбор способов реализации функций сетевых протоколов (аппаратная реализация, программная реализация, применение проблемно ориентированных интерфейсных СБИС или сопроцессоров)
- разработка инструментальных средств автоматизации программирования (языки программирования, компиляторы, ассемблеры, редакторы связей, загрузчики, библиотеки функций).

### **Выводы**

Сетевые процессоры являются технологией, направленной на быструю разработку гибких и легко модифицируемых аппаратных решений для компьютерных сетей.

Целью этой технологии является создание СБИС для обработки пакетов, передаваемых по сетям, которые эффективно сочетают гибкость программируемых процессоров с высоким быстродействием проблемно-ориентированных СБИС.

В настоящее время накоплен определенный опыт проектирования и эксплуатации сетевых процессоров. Однако отсутствуют ясность в том, какие функции обработки пакетов должны быть реализованы в сетевых процессорах. Не определены наиболее эффективные аппаратные архитектуры сетевых процессоров. Параметры применяемых в выпускаемых промышленностью сетевых процессорах комбинированных параллельно - конвейерных структур не всегда имеют теоретическое обоснование, не удовлетворяют постоянно возрастающим требованиям по производительности.

Разработка программного обеспечения для сетевых процессоров может представлять серьезную проблему, поскольку для обеспечения высокой эффективности многие сетевые процессоры нуждаются в низкоуровневом микропрограммировании.

### **Литература**

1. Олифер В.Г., Олифер Н.А. Основы компьютерных сетей. – СПб.: Питер, 2009.-352с.
2. Кучерявый Е.А. Управление трафиком и качество обслуживания в сети Интернет. – СПб.: Наука и Техника, 2004. – 336с.
3. Network Processors: Programmable Technology for Building Network Systems  
[http://www.cisco.com/web/about/ac123/ac147/archived\\_issues/ipj\\_7-4/network\\_processors.html](http://www.cisco.com/web/about/ac123/ac147/archived_issues/ipj_7-4/network_processors.html)
4. Орлов С.А., Цилькер Б.Я. Организация ЭВМ и систем. 2-е изд. – СПб.: Питер, 2011.-688с.