

п р и к л а д н а я

ИНФОРМАТИК@



научно-практический
журнал

№ 2 (38) 2012

Март-апрель

ISSN 1993-8314

С 19 февраля 2010 года журнал включен в Перечень ведущих периодических изданий, рекомендованных ВАК для публикации результатов диссертационных исследований.

Уважаемые коллеги!

В конце апреля завершает свою работу VII Международный научный конгресс «Роль бизнеса в трансформации российского общества — 2012». С целями и задачами этого форума можно ознакомиться на второй странице обложки журнала.

Вслед за конгрессом нас ожидают два очень интересных события, связанных с внедрением высокоинтеллектуальных методов и соответствующего «интеллектного» программного обеспечения в современные информационные системы поддержки принятия решений.

Во-первых, 27 апреля в Санкт-Петербурге состоится собрание Национального общества имитационного моделирования России (НОИМ) по актуальным вопросам (Санкт-Петербургский институт информатизации и автоматизации РАН). В состав Правления этого общества входят главный редактор и некоторые члены редакционного совета нашего журнала. Национальное общество является одним из главных организаторов российских и международных конференций по прикладной информатике. Во-вторых, в Москве 15 мая пройдет Всероссийская научно-практическая конференция по имитационному моделированию социально-экономических систем (ВЗФЭИ). Фактически это продолжение грандиозного форума ИММОД-2011 в Санкт-Петербурге, состоявшегося в октябре прошлого года. Журнал «Прикладная информатика» — один из организаторов и спонсоров московской конференции, как и на ИММОД-2011. На майском форуме планируются заседания следующих секций:

- теоретические основы и методология имитационного моделирования;
- методы исследования больших экономических систем;
- моделирование глобальных процессов;
- практическое применение программных средств;
- имитационное моделирование в обучении и образовании.

Редакционный совет журнала желает своим читателям, авторам и подписчикам плодотворной творческой работы наступившей весной!

Главный редактор
А. А. Емельянов

Редакционный совет

Главный редактор

Емельянов А. А., докт. экон. н., проф., вице-президент МФПУ «Синергия», зав. кафедрой Математических и инструментальных методов экономики

Сопредседатели редакционного совета

Рубин Ю. Б., докт. экон. н., проф., чл.-корр. РАО, ректор МФПУ «Синергия», зав. кафедрой Теории и практики конкуренции

Мешалкин В. П., докт. техн. н., проф., чл.-корр. РАН, директор Института логистики ресурсосбережения и технологической инноватики, зав. кафедрой Логистики и экономической информатики РХТУ им. Д. И. Менделеева

Члены редакционного совета

Амбросов Н. В., докт. экон. н., проф., зав. кафедрой Информатики и кибернетики БГУЭФ (Иркутск)

Бендиков М. А., докт. экон. н., проф., зав. кафедрой Инновационного управления и моделирования МФПУ «Синергия», ведущий научный сотрудник ЦЭМИ РАН

Бугорский В. Н., канд. экон. н., проф. СПбГИЭУ (ИНЖЭКОН)

Волкова В. Н., докт. экон. н., проф. СПбГПУ

Дуго С. М., канд. экон. н., проф., Компания «1С», отв. за работу с авторизованными учебными центрами и образовательными учреждениями

Дик В. В., докт. экон. н., проф., зав. кафедрой Информационного менеджмента и электронной коммерции МФПУ «Синергия»

Дли М. И., докт. техн. н., проф. филиала МЭИ (ТУ) в Смоленске, зав. кафедрой Менеджмента и информационных технологий в экономике

Козлов В. Н., докт. техн. н., проф., зав. кафедрой Системного анализа и управления СПбГПУ

Коршунов С. В., канд. техн. н., проф., проректор МГТУ им. Н. Э. Баумана

Мэйпл Карстен, Ph. D., проф., глава Департамента Прикладных вычислений Бэдфордширского университета (Великобритания)

Павловский Ю. Н., докт. физ.-мат. н., проф., чл.-корр. РАН, Вычислительный центр им. А. А. Дородницына РАН, зав. отделом Имитационных систем

Потемкин А. И., докт. техн. н., проф. РГУТиС

Росс Г. В., докт. экон. н., докт. техн. н., проф., заместитель директора ВНИИ ПВТИ

Салмин С. П., докт. экон. н., проф. МФПУ «Синергия»

Саркисов П. Д., докт. техн. н., академик РАН, президент РХТУ им. Д. И. Менделеева

Сухомлин В. А., докт. техн. н., проф., директор Центра IT-образования МГУ им. М. В. Ломоносова

Халин В. Г., докт. экон. н., проф., зав. кафедрой Информационных систем в экономике СПбГУ

Хубаев Г. Н., докт. экон. н., проф., зав. кафедрой Экономической информатики и автоматизации управления РГЭУ (РИНХ, Ростов)

Чистов Д. В., докт. экон. н., проф., зав. кафедрой Информационных технологий Финансового университета при Правительстве РФ

Шориков А. Ф., докт. физ.-мат. н., проф., ведущий научный сотрудник Центра экономической безопасности Института экономики Уральского отделения РАН (г. Екатеринбург)

Заместители главного редактора

Власова Е. А., научная редакция Редакционно-издательского центра МФПУ «Синергия»

Прокимов Н. Н., канд. техн. н., доцент кафедры Математических и инструментальных методов экономики МФПУ «Синергия»

Editorial Board

Editor in Chief

A. Emelyanov, Doctor of Economics, Professor, Moscow University of Finance and Industry "Sinergy", Vice-President, Head of the Mathematical and Instrumental Methods of Economics Department

Co-Chairs of the Editorial Board

Yu. Rubin, Doctor of Economics, Professor, Corresponding Member of the Russian Education Academy, Moscow University of Finance and Industry "Sinergy", Rector, Head of the Theory and Practice of Competition Department

V. Meshalkin, Doctor of Technical Sciences, Professor, Corresponding Member of Russian Academy of Sciences, Mendeleyev University of Chemical Technology of Russia, Director of the Institute of Logistics and Resource Technology Innovation, Head of the Logistics and Economical Informatics Department

Members of the Editorial Board

N. Ambrosov, Doctor of Economics, Professor, Baikal State University of Economics and Law (Irkutsk), Head of the Informatics and Cybernetics Department

M. Bendikov, Doctor of Economics, Professor, Moscow University of Finance and Industry "Sinergy", Head of the Innovation Management and Modeling Department, leading researcher of the Central Institute of Mathematical Economics RAS

V. Bugorskiy, PhD, Professor, St. Petersburg University of Engineering and Economics, the Economic Information Systems Department

M. Carsten, PhD, Professor, University of Bedfordshire, United Kingdom, Head of the Applicable Computing Department

D. Chistov, Doctor of Economics, Professor, University of Finance Government of the Russian Federation, Head of the Information Technology Department

V. Dick, Doctor of Economics, Professor, Moscow University of Finance and Industry "Sinergy", Head of the Information Management and Electronic Commerce Department

S. Digo, PhD, Professor, Company «1С», Account Manager, Authorized Training Centers and Educational Institutions

M. Dli, Doctor of Technical Sciences, Professor, Branch of the Moscow Power Engineering Institute (Technical University) in Smolensk, Head of the Management and Information Technology in the Economy Department

V. Hulin, Doctor of Economics, Professor, St. Petersburg State University, Head of the Economic Information Systems Department

G. Khubayev, Doctor of Economics, Professor, Rostov State Economic University (Rostov), Head of the Economic Informatics and Automation Control Department

S. Korshunov, PhD, Professor, Bauman Moscow State Technical University, vice-rector

V. Kozlov, Doctor of Technical Sciences, Professor, St. Petersburg State Polytechnic University, Head of the System Analysis and Control Department

Y. Pavlovsky, Physical and Mathematical Sciences, Professor, Corresponding Member of Russian Academy of Sciences, the Dorodnitsyn Computing Centre of RAS, Head of Simulation Systems Department

A. Potemkin, Doctor of Technical Sciences, Professor, Russian State University of Tourism and Service, Head of the Corporate Governance and E-Business Department

G. Ross, Doctor of Economics, Doctor of Technical Sciences, Professor, Deputy Director of All-Russian Research Institute for Problems of Computer Technology and Information

S. Salmin, Doctor of Economics, Professor, Moscow University of Finance and Industry, the Mathematical and Instrumental Methods of Economics Department

P. Sarkisov, Doctor of Technical Sciences, Professor, Academician of Russian Academy of Sciences, D. Mendeleyev University of Chemical Technology of Russia, President

A. Shorikov, Doctor of Physical and Mathematical Sciences, Professor, Senior Researcher, Economic Security Center, Institute of Economy, Ural Department of Russian Academy of Science (Ekaterinburg)

V. Sukhomlin, Doctor of Technical Sciences, Professor, Lomonosov Moscow State University, Director of the IT-Education Center

V. Volkova, Doctor of Economics, Professor, St. Petersburg State Polytechnic University, the Economic Information Systems Department

Deputy Chief Editors

E. Vlasova, Moscow University of Finance and Industry "Sinergy", Scientific Edition Department

N. Prokimov, PhD, Associate Professor, Moscow University of Finance and Industry "Sinergy", the Mathematical and Instrumental Methods of Economics Department

IT-бизнес

Прогнозирование рынка

Ю. В. Трифионов, Е. С. Митяков, В. А. Сазонтов
 Фрактальный анализ поведения быстрых индикаторов экономики России 5

Рынок технологий

А. А. Салтан
 Моделирование рынка программного обеспечения при наличии внешнего сетевого эффекта и компьютерного пиратства..... 9

IT-менеджмент

Управление ресурсами

М. А. Телемтаев, А. А. Попов
 Совершенствование отечественных информационных систем управления недвижимостью на основе зарубежного опыта..... 18

Управление эффективностью

Р. Е. Кузин, О. В. Кожин, И. В. Лебедев, А. М. Могирев, С. С. Писаненко, Т. Н. Таиров,
 Информационная поддержка системы радиационного контроля при большом потоке исследуемых проб..... 26

Автоматизированные системы управления

В. Ф. Корнюшко, М. М. Кулыгина, В. М. Норица, И. В. Храпов
 Информационная поддержка мониторинга государственных закупок..... 32

IT и образование

Подготовка IT-специалистов

М. В. Слива
 Кроссплатформенный подход как средство унификации обучения программированию в различных операционных системах..... 38

Образовательное пространство

Е. В. Любимов, Г. П. Озерова
 Системный подход к управлению информатизацией университета..... 46

Н. А. Стефанова

Оценка эффективности поисковой оптимизации сайтов вузов с использованием поисковых запросов..... 55

Инструментальные средства

Технология разработки программного обеспечения

А. М. Пеленицын
 Методы обобщенного и метапрограммирования в программной реализации декодера алгебро-геометрических кодов..... 60

Сетевые технологии

Информационная безопасность

П. Б. Филиппов
 Использование и реализация защиты персональных данных в социальных сетях Интернета..... 71

3D моделирование

Виртуальная реальность

А. И. Разумовский
 Практика создания 3D-тренажеров..... 78

Математический аппарат

А. В. Кучуганов, Д. Р. Касимов
 Графический поиск чертежей в хранилищах данных..... 84

Simulation

Акторное моделирование

А. А. Емельянов, Е. А. Власова, Н. З. Емельянова, Н. Н. Прокимнов,
 Имитационное моделирование инвестиционных процессов..... 93

Лаборатория

Моделирование информационных процессов

С. И. Маторин, О. А. Зимовец
 Формализованное визуальное моделирование административных процедур.... 100

Управление производством

Г. Н. Хубаев
 Оценка резервов снижения ресурсоемкости товаров и услуг: методы и инструментальные средства..... 111

Системы поддержки принятия решений

С. В. Астанин, Н. К. Жуковская
 Внутрифирменные механизмы распределения ограниченных ресурсов на основе переговорного процесса..... 118

Вопросы теории

Математические методы

В. Е. Кацман
 Модель формирования оптимальной структуры аппарата управления организации.... 125

Сведения об авторах..... 130

Аннотированный список статей..... 134

Правила оформления рукописей..... 139

IT business

Market's prediction

Yu. Trifonov, E. Mityakov, V. Sazontov

Fractal analysis of Russian economy
fast indicators behavior5

Technology market

A. Saltan

Software market modeling in the network
externalities and digital piracy environment9

IT Management

Resource management

M. Telemtaev, A. Popov

Using international experience for improving
domestic information systems of property
management18

Performance management

*R. Kuzin, O. Kozhin, I. Lebedev,
A. Mogirev, S. Pisanenko, T. Tairov*

Information support of radiation control system
at a high flow of samples26

Automated control systems

V. Korniyushko, M. Kulygina, V. Noritsa, I. Khrapov

Information support for public
procurement monitoring32

IT and education

Training IT specialists

M. Sliva

Cross-platform approach as a means of teaching
programming unification in a context of various
operational systems38

Educational environment

E. Lyubimov, G. Ozerova,

Using system approach to the University
computerization management46

N. Stefanova

Indices of Search Queries55

Tools

Software engineering

A. Pelenitsyn

Generic and meta- programming methods
in software implementation of decoder
of algebraic geometry codes60

Network technologies

Information security

P. Filippov

Using and protecting of personal data
in Internet social networks71

3D modeling

Virtual reality

A. Razumovsky

The practice of creating 3D simulators78

Mathematical tools

A. Kuchuganov, D. Kasimov

Graphical search of technical drawings
over data warehouses84

Simulation

Actor modeling

*A. Emelianov, E. Vlasova,
N. Emelianova, N. Prkkinov*

Simulating investment processes93

Laboratory

Information processes modeling

I. Matorin, O. Zimovets

Formalized visual modeling
of administrative procedures100

Production management

G. Khubaev

Evaluating the reserves to reduce
resource consumption for producing goods
and services: methods and tools111

Decision support systems

S. Astanin, N. Zhukovskaya

Tools of enterprise finite resources distribution
on the basis of negotiation118

Theoretical approach

Mathematical methods

V. Katsman

The model of constructing the optimal
organization management structure125

Authors130

Abstracts134

Guidelines for authors139

А. М. Пеленицын, аспирант Южного федерального университета, Ростов-на-Дону

Методы обобщенного и метапрограммирования в программной реализации декодера алгебро-геометрических кодов

Компьютерное решение ряда задач, в частности задач теории кодирования, может представлять интерес с точки зрения не только полученного результата, но также и применяемых для решения методов.

Введение

Конструкции помехоустойчивых кодов на основе алгебраических кривых (АГ-кодов) и методы их декодирования [1] используют ряд абстрактных математических объектов, программная реализация которых представляет собой более сложную задачу, чем в случае классических кодов. Учитывая тот факт, что «открытых» реализаций кодеков для АГ-кодов почти нет или мало [2, 3], опыт их реализации представляет интерес. Увеличение числа программных пакетов такого типа может также служить накоплению экспериментальных данных в области помехоустойчивого кодирования.

Помимо интереса с теоретико-кодовой точки зрения компьютерное решение ряда алгебраических задач, в том числе из теории кодирования, ставит специфические задачи в программной инженерии. Одним из важных вопросов в таких задачах является выразительность систем типов используемых языков программирования. Было замечено (например, в [4, 3.1]), что система типов языка C++, а именно, методы, основанные на механизме шаблонов, в том числе метапрограммирование, позволяют увеличить выразительность создаваемого программного кода по сравнению с общепринятыми методами объектно-ориентированного

и, тем более, процедурного программирования.

В алгебраических задачах, как правило, приходится иметь дело с рядом абстрактных структур, каждая из которых может иметь различные «носители». Например, реализация арифметики в простых конечных полях отличается от реализации в случае конечных полей, являющихся расширениями простых. Однако с точки зрения алгоритмов над конечными полями это различие не играет никакой роли и по возможности должно быть скрыто за (обобщенными) интерфейсами, что позволит избежать дублирования кода. При этом необходимо сохранять максимальную эффективность. Именно такая задача является почти дословной формулировкой принципа обобщенного программирования Мёссера–Степанова [5] (реализация алгоритмов наиболее независимым от представления входных данных образом). В ряде работ было замечено, что для ее решения, а значит, и для воплощения принципа обобщенного программирования при реализации линейно-алгебраических и графовых алгоритмов, удобно использовать уже упомянутый механизм шаблонов C++ [6, 7, 8]. Данную статью можно считать продолжением указанных исследований применительно к аппарату вычислительной алгебраической геометрии [9, chap. 10].

Статья состоит из введения, двух разделов и заключения. В первом разделе прово-

дится обзор полученной реализации декодера. Это сделано для того, чтобы во втором разделе при описании приемов обобщенного и мета-программирования была лучше ясна их роль в общем дизайне программной системы. Кроме того, обзор реализации дает представление о роли обобщенных интерфейсов в проектировании программных систем. Первый раздел имеет четыре подраздела, соответствующих основным модулям системы. Второй раздел описывает наиболее интересные приемы обобщенного и метапрограммирования, примененные в реализации отдельных элементов данной программной системы. Второй раздел также состоит из четырех подразделов, соответствующих наиболее интересным приемам, выбранным для рассмотрения в статье.

1. Обзор реализации

Создана библиотека, реализующая работу с полиномами многих переменных в объеме, необходимом для выполнения BMS-алгоритма, который является *основным конструктивным звеном* в декодирующем алгоритме, а также сам BMS-алгоритм и алгоритм декодирования на его основе, описанный в [9, chap. 10]. В реализацию не включена процедура голосования большинством Фенга и Рао, что несколько снижает декодирующую силу алгоритма, но не снижает значимости полученных в статье результатов. Сообщение о реализации BMS-алгоритма сделано в [10], в настоящей статье более подробно рассматриваются, в том числе, приемы, использованные в этой реализации, но не описанные в [10].

При реализации библиотеки использовался язык программирования C++, включая инструменты, появившиеся в его новом стандарте [11] (чтобы особо отметить использование этих инструментов, ниже будет употребляться обозначение «C++11»). В связи с распространением компиляторов, в основном реализующих новый стандарт (несколько последних версий компилятора из набора GCC и версия компилятора Visual

C++ 2010 г.), это не должно накладывать существенных ограничений на использование данной программной реализации.

Библиотека в терминологии C++ является «headers-only», т.е. вся реализация содержится в заголовочных файлах C++, предназначенных для включения в виде текста в использующие их проекты, и, таким образом, существует, прежде всего, в виде открытых исходных кодов. Наиболее важны четыре шаблона классов:

```
– Point<N, OrderPolicy>,
– Polynomial<T>,
– BMSAlgorithm< SeqT, PolynomialT,
OrderPolicy >,
– BMSDecoding<N, ECCCodeParams>
```

(часть параметров шаблонов здесь и далее для краткости опускается). Кроме того, имеется набор вспомогательных шаблонов классов и функций, занимающих в общей сложности около 4 тыс. строк кода вместе с включенной в исходный код документацией. Наличие свободных функций является отступлением от чистого объектно-ориентированного дизайна и следует рекомендациям [12, п. 44], касающимся использования языка программирования C++ как мультипарадигменного. Также, в соответствии с последним, в данной реализации особое внимание уделено применению обобщенного программирования на основе шаблонов C++. Объектно-ориентированный подход используется лишь в малой степени как дополнительное средство поддержки модульности; наследование применяется лишь в нескольких местах для удобства реализации идиом обобщенного программирования; не используется (динамический, основанный на виртуальных функциях) полиморфизм.

1.1. Шаблон класса Point<N>

Шаблон класса Point<N> моделирует точку в N-мерной целочисленной решетке. С точки зрения реализации он представляет собой обертку над шаблоном `std::array<N, int>` из нового стандарта библиотеки C++, который, в свою очередь, является стати-

ческим массивом с STL-совместимым интерфейсом. К отдельным координатам точки можно обращаться с помощью операции взятия индекса (`operator []`). Важной частью интерфейса `Point<N>` являются функции, реализующие различные упорядочения на множестве точек решетки, такие как `byCoordinateLess` и `totalLess` или его синоним `operator<`. Первая (свободная функция) обеспечивает покоординатное сравнение точек (частичное упорядочение), вторая (функция-член) задает специальное мономиальное упорядочение [9, с. 7], которое, в частности, является полным порядком, что позволяет хранить объекты `Point<N>` в упорядоченных ассоциативных контейнерах стандартной библиотеки C++ (например, в качестве ключей шаблона класса `std::map`). Кроме того, имеется набор свободных функций, реализующих поиск в наборе точек минимумов и максимумов относительно разных порядков, поиск в наборе точек точки меньше заданной, и т.п. Все это необходимо выполнять на разных стадиях BMS-алгоритма.

Реализация `totalLess` в форме функции-члена (в отличие от других функций, связанных с упорядочением) связана с желанием параметризовать тип точки мономиальным упорядочением. Такая параметризация доступна с помощью использования второго параметра шаблона `Point<N>`, который имеет значение по умолчанию, соответствующее градуированному антилексикографическому упорядочению. Этот прием будет подробно описан в следующем разделе.

1.2. Шаблон класса `Polynomial<T>`

Шаблон класса `Polynomial<T>` представляет тип полинома, параметр типа `T` обозначает тип коэффициентов соответствующего множества полиномов. Для получения полиномов многих переменных используется свойство, позволяющее представлять полиномы полиномами меньшей степени (см., к примеру, [13, гл. IV, § 1, п. 5]). Например, полином от двух переменных с коэффициентами типа `T` неотличим по алгебраическим

свойствам от полинома от одной переменной с коэффициентами, представляющими собой полиномы от одной переменной с коэффициентами типа `T`. Таким объектам соответствует тип `Polynomial<Polynomial<T>>`.

Для удобства использования полиномов многих переменных введен отдельный шаблон класса `MVPolyType<n, T>`, где первый параметр указывает на количество переменных в результирующем типе. Например, следующие две строки кода определяют переменные одинакового типа, представляющего полином от трех переменных с целочисленными коэффициентами.

```
Polynomial<Polynomial
< Polynomial<int> > > p;
MVPolyType<3, int>::ResultT q;
```

Полиномы можно задавать в специальном строковом представлении, перечисляя коэффициенты полинома в квадратных скобках через пробел в порядке возрастания степеней. Например, “[2 0 1]” соответствует полиному $x^2 + 2$. Так как полином от нескольких переменных есть обычный полином с коэффициентами — полиномами от переменных, число которых меньше на единицу числа переменных данного полинома, то строковое представление таких полиномов будет содержать вложенные скобки. Например, “[[2 0 2] [1] [0 3]]” соответствует полиному $3x^2y + 2y^2 + x + 2$ (с точностью до возможного переименования переменных).

Для полиномиальных типов перегружены арифметические операции, которые необходимы для реализации BMS-алгоритма и декодера. Это сложение полиномов, умножение полинома на скаляр, умножение полинома на моном (`operator<<`) и вычисление полинома в точке. Таким образом, из основных операций с полиномами не реализовано лишь умножение. Продемонстрируем использование операции умножения на моном. Считается, что моном задается своей степенью, которая, в свою очередь, представлена типом `Point<N>`. Пример умножения полинома от двух переменных на моном x^3y^2 :

```
Point<2> mon;
mon [0] = 3;
mon [1] = 2;
p <<= mon;
```

Стоит отметить, что, так как шаблоны и перегрузка операций гарантируют контроль типов времени компиляции, то замена в данном примере двоек в аргументах шаблонов на два произвольных несовпадающих числа привела бы к ошибке компиляции.

Примером того, как элементы C++11 упрощают решение ряда типичных задач, является агрегатная инициализация объектов `Point<N>` — код с ее использованием, эквивалентный приведенному выше, выглядит так:

```
Point<2> mon {3, 2};
p <<= mon;
```

Чтобы такой синтаксис создания объекта класса стал доступен, необходимо было определить конструктор с параметром типа `std::initializer_list<T>`, где `T` указывает ожидаемый тип элементов в фигурных скобках.

Для полиномов многих переменных нельзя однозначно («наилучшим образом») определить степень. Для выполнения BMS-алгоритма требуется задавать некоторое мономиальное упорядочение на множестве мономов, и определить степень возможно. Однако работа со степенями полиномов, получаемых на каждой итерации алгоритма, ведется в значительной мере независимо от самих полиномов: на каждой итерации сначала определяется геометрический образ очередного приближения к решению, составленный из степеней будущих полиномов в N -мерной решетке, а затем для каждой степени строится полином. Указанный подход обусловил следующее решение: в данной реализации тип полинома не хранит никакой информации о своей степени (что бы ни подразумевалось под степенью полинома от нескольких полиномов), если такая информация нужна клиенту типа, он должен хранить и обновлять ее самостоятельно.

1.3. Шаблон класса `BMSAlgorithm`

Шаблон класса `BMSAlgorithm<PolynomialT>` параметризован типом полиномов (в частности, алгоритм может работать с полиномами различного числа переменных) и имеет довольно простой интерфейс, отражающий назначение и использование BMS-алгоритма. Напомним, что BMS-алгоритм по заданной финитной n -мерной последовательности над конечным полем $\mathbf{N}^n \rightarrow \mathbf{F}_q$ строит конечное множество полиномов от n переменных, называемое *минимальным множеством* этой последовательности. Интерфейс, который должна поддерживать последовательность, подаваемая на вход алгоритму, предельно сжат. Большая часть операций, проводимых с последовательностью в ходе алгоритма, аналогична операциям, определенным для типа многочленов. Конструктор класса принимает последовательность и точку, обозначающую конец последовательности, алгоритм обрабатывает все элементы в диапазоне от элемента с мультииндексом $(0, \dots, 0)$ до этого маркера конца последовательности в порядке, заданном используемым мономиальным упорядочением (напомним, что конкретное упорядочение — часть описания типа точки). Метод данного шаблона класса `computeMinimalSet` возвращает коллекцию с STL-совместимым интерфейсом, содержащую выход алгоритма. Тип этой коллекции является частью определения шаблона класса, к нему можно обратиться следующим образом:

```
BMSAlgorithm<PolynomialT>::PolynomialCollection
```

1.4. Шаблон класса `BMSDecoding`

Интерфейс типа `BMSDecoding` достаточно лаконичен: в конструкторе задается параметр l кода C_l [9, chap. 10], единственная открытая функция-член `decode` принимает вектор, пришедший по каналу, представленный типом `BMSDecoding::FieldElmsCollection` с STL-совместимым интерфейсом (в текущей реализации это `std::vector`).

2. Некоторые приемы обобщенного и метапрограммирования в реализации декодера одного класса алгебро-геометрических кодов

2.1. Рекурсивное инстанцирование шаблонов

Важным проектным решением, опирающимся на особенности шаблонов C++, является рекурсивное определение полиномов многих переменных: `Polynomial<... Polynomial<T>...>`, где глубина вложенности равна n , для моделирования полинома от n переменных. Идея типа, параметризованного самим собой, так называемое «рекурсивное инстанцирование» шаблонов классов, является известным шаблоном проектирования [14, гл. 21]. Однако приемы обработки такого рода структур все еще недостаточно развиты.

При появлении рекурсивной структуры возникает вопрос о способе ее обхода. В разных ситуациях это может потребовать более или менее сложных шаблонных техник. Приведем некоторые из них. Заметим, что часть примеров касается перегруженных операций для полиномов, и поскольку в реализации использована «каноническая форма» [12, п. 27] арифметических операций, вся реализация помещена в присваивающие версии операций, которые реализованы как функции-члены шаблона класса `Polynomial<T>`.

При создании шаблонов классов с использованием рекурсивного инстанцирования обычно предпринимаются действия для облегчения их использования. В данном случае для этой цели создан шаблон `MVPolyType<n, T>`. Его определение дает представление о типичном подходе к работе с рекурсивно инстанцируемыми типами.

```
template<int VarCnt, typename Coef>
struct MVPolyType {
    typedef Polynomial<
// "recursive call" to MVPolyType:
        typename MVPolyType<VarCnt - 1,
Coef>::ResultT >
        ResultT;
};
template<typename Coef>
```

```
struct MVPolyType<1, Coef> {
    typedef Polynomial<Coef> ResultT;
};
```

Как можно видеть из приведенного текста, рекурсия проводится по количеству переменных полинома, для полинома одной переменной определена специализация шаблона, которая отвечает за остановку рекурсии. Рекурсия по типу, таким образом, не сильно отличается от простейших алгоритмов, использующих рекурсию по данным. На этом примере также видно, как с помощью использования механизма шаблонов на этапе компиляции можно генерировать новые программные сущности — типы многочленов от разного числа переменных с разными типами коэффициентов, — что является одним из вариантов реализации концепции метапрограммирования.

Аналогичным образом внутри типа полинома `Polynomial<T>` задается константа `VAR_CNT`, определяющая количество переменных полинома в зависимости от типа `T`, и синоним типа `CoefT` для обозначения фактического типа коэффициентов полинома в случае нескольких переменных. Например,

```
Polynomial<Polynomial<int>>::VAR_
CNT
равно 2 и Polynomial<Polynomial<int>
>::CoefT
обозначает int.
```

Изложенное выше связано с обходом и получением информации о самом типе, имеющем рекурсивное определение. Далее рассматривается работа с объектами такого типа.

Операция умножения полинома на скаляр не требует никаких дополнительных усилий по сравнению со случаем полинома от одной переменной.

```
Polynomial operator *=(CoefT const & c )
{
    for (typename StorageT::iterator
it = data.begin();
it != data.end(); ++it)
    {
        (*it) *= c;
    }
}
```

```
return *this;  
}
```

Тип полинома является, по сути, контейнером особого рода и реализован на базе одного из стандартных контейнеров, который скрыт за синонимом типа `StorageT`. Соответствующее поле шаблона класса называется `data`.

Под умножением на скаляр понимается вызов для полинома операции `*` с объектом, тип которого совпадает с `CoefT`. Непосредственно умножение производится вызовом той же операции для каждого элемента контейнера `data`. Если мы имеем дело с полиномом одной переменной, то `data` должен содержать элементы типа `CoefT`, для которого, таким образом, должна быть определена операция `*` (тип коэффициентов полинома должен допускать умножение). В ином случае (`data` содержит элементы другой инстанции `Polynomial`) снова будет вызвана приведенная выше функция, однако изменится тип `Polynomial` — теперь это будет тип полинома от переменных, число которых меньше на единицу. Еще раз подчеркнем, что описанная логика полностью реализована в приведенном выше коде, поэтому не требуется, к примеру, использовать специализацию шаблонов, чтобы явно останавливать рекурсию.

Программный код сложения двух полиномов так же, как умножения на скаляр, не требует дополнительных модификаций, учитывающих использование рекурсивного инстанцирования. Можно сделать вывод о том, что такого рода рекурсивный обход одного (умножение на скаляр) или нескольких (сложение, сравнение двух полиномов на равенство) объектов одного рекурсивно инстанцированного типа может быть довольно удобен. Однако при работе с полиномами многих переменных возникает задача рекурсивного обхода объектов двух разных типов одновременно (при этом глубина рекурсии разных типов должна быть одинаковой). Примерами решения такой задачи в реализации данной библиотеки являются операции умножения полинома на мо-

ном и обращения к коэффициенту полинома от многих переменных по его мультииндексу (иначе говоря, по мультистепени монома, при котором стоит запрашиваемый коэффициент). В обоих случаях требуется осуществлять одновременный обход объекта типа `Polynomial<...Polynomial<T>...>` и объекта типа `Point<N>`, где число `N` совпадает с глубиной вложенности типа полинома (а значит, в соответствии с принципами библиотеки, с числом переменных полинома).

Рассмотрим реализацию операции обращения к коэффициенту полинома по его (мульти) индексу. В частном случае, при работе с полиномом от одной переменной обращаться к коэффициенту хотелось бы по обычному целочисленному индексу, по причине чего реализована версия `operator [] (int)`. Для обращения к коэффициенту полинома от многих переменных предоставляется перегруженная версия `operator [] (Point<VAR_CNT>)` (размерность точки должна совпадать с количеством переменных полинома). Обсудим вопрос о том, как должна продвигаться рекурсия.

Пусть задана точка `p` типа `Point<VAR_CNT>`. Для полинома от многих переменных нужно обратиться к `p [0]`-му элементу в контейнере `data` (это полином от переменных, количество которых меньше на одну) и снова вызвать для него `operator [] (Point<VAR_CNT>)` с аргументом, представляющим как бы срез точки `p`: точку со всеми элементами `p`, кроме `p [0]`-го. Для этой цели был создан шаблон класса `Slice<Dim, Offset>`, который хранит ссылку на исходную точку размерности `Dim` и поддерживает операцию взятия индекса, причем при обращении к `i`-му элементу по индексу возвращается `Offset+i`-й элемент исходной точки. Используя этот шаблон в реализации `operator [] (Point<VAR_CNT>)` хотелось бы написать что-то наподобие `(data [p [0]]) [make_slice (p)]`, где функция `make_slice` создает срез, исключаяющий первый элемент точки. При этом в интерфейс полинома нужно добавить `operator [] (Slice<Dim, Offset>)`, в котором будет написано примерно то же,

что и в исходной версии этой операции, имеющей в качестве аргумента точку. К сожалению, точно так написать нельзя по причине того, что в таком варианте не получается остановить рекурсию.

Для остановки рекурсии в шаблонных типах обычно используется специализация шаблона (как в примере с `MVPolyType`). В данном случае нужно было бы создать специализацию `operator [] (Slice<Dim, Dim-1>)`. Однако в языке C++ [14, п. 12.3.3] запрещается специализировать шаблонные члены шаблонов класса (именно таким является `operator [] (Slice<Dim, Offset>)`). Так появляется еще один уровень косвенности, вместо (`data [p [0]]`) [`make_slice (p)`] используется вызов отдельной свободной функции `apply_subscript (data [p [0]], make_slice (p))`, которая либо продвигает рекурсию, вызывая `operator [] (Slice<Dim, Offset>)` с переданным ей срезом, либо (в своей специализации) останавливает рекурсию, вызывая `operator [] (int)` с нулевым элементом полученного среза (этот элемент стоит в последней координате исходной точки).

Удобно считать, что при обращении по индексу, выходящему за пределы текущего контейнера, нужно вернуть «нулевое значение». Такое значение получается с помощью шаблона `CoefficientTraits`. Итоговый код, реализующий операцию обращения по индексу, выглядит так:

```
template<typename T, typename S,
typename Pt>
T applySubscript(S const & el, Pt
const & pt) {
    return el[pt];
// call for operator[] (Slice<...>)
}
template<typename T, typename S, int
Dim>
// recursion stop
T applySubscript(S const & el,
Slice<Dim, Dim-1> const & pt {
    return el[pt[0]];
// call for operator[] (int)
}
template<typename T>
```

```
typename Polynomial<T>::CoefT
Polynomial<T>::operator[] (Point<VAR_
CNT> const & pt) const {
if (pt[0] < 0 || data.size() <= pt[0])
return CoefficientTraits<CoefT>::addI
d();
else
return
applySubscript<Polynomial<T>::CoefT>(d
ata[pt[0]],
make_slice(pt));
}
```

Первая из приведенных версий `applySubscript`, продвигая рекурсию, вынуждена работать с все новыми экземплярами шаблона `Slice`, но поскольку конкретные значения шаблонных параметров `Slice` не важны для нее, то это дает основания считать применяемый подход подходом, основанным на принципах обобщенного программирования: алгоритм сделан максимально независимым от представления данных.

Код `operator [] (Slice<...>)` не приводится, так как он полностью повторяет `operator [] (Point<VAR_CNT>)`, а `operator [] (int)` отличается только тем, что вместо вызова `apply_subscript` стоит `data [pt]` (`pt` — полученное целое число).

Приведенный код, осуществляющий обход рекурсивной структуры типа полинома, является довольно громоздким. Поскольку реализация еще нескольких операций (например, вывода полинома в поток и вычисления полинома для некоторого значения переменных) осуществлена аналогично, то возникает вопрос, можно ли создать более общую обертку для реализации операции обхода рекурсивной структуры параметризованную действиями, которые следует выполнять при обходе. На данном этапе ответ на этот вопрос найден не был.

2.2. Характеристики типов и проблема дублирования в них кода

Упомянутый выше шаблон `CoefficientTraits` представляет собой пример шаблона класса характеристик, или, в другом переводе, свойств типов (`type traits`) [14, гл. 15],

[15, п. 2.10]. В данном случае этот класс предоставляет необходимую полиному информацию о типе коэффициентов, а именно, значение этого типа, играющее роль аддитивно нейтрального элемента (см. пример выше), мультипликативно нейтрального:

```
CoefficientTraits<T>::multId ()
```

функцию, возвращающую аддитивно обратный элемент по заданному:

```
CoefficientTraits<T>::addInverse (T)
```

и мультипликативно нейтральный:

```
CoefficientTraits<T>::multInverse (T)
```

Применение характеристик основано на использовании специализаций шаблонов. К примеру, если клиент шаблона класса полинома инстанцирует его с типом `Var` для коэффициентов, нулевое значение которого хранится в статической константе `Var::ZERO`, то клиенту нужно предоставить специализацию `CoefficientTraits`, функция `addId` которого возвращает `Var::ZERO`. Имеется общее определение `CoefficientTraits<T>`, которое будет использовано, если клиент не сделает последнего, оно реализует наиболее принятое поведение (например, нулевое значение создается с помощью вызова конструктора без параметров типа `T`).

При описании характеристик типов обычно упускают вопрос: как избежать дублирования в разных инстанциях шаблона характеристик. Он представляется достаточно общим, в данном случае он возникает из-за того, что библиотека NTL [16], которая предоставляет реализацию арифметики в конечных полях, имеет четыре класса для полей разного типа: простых полей характеристики, отличной от двух, расширенных полей характеристики, отличной от двух, F_2 и его расширений. При этом все четыре класса имеют почти идентичный интерфейс, поэтому при создании специализаций `CoefficientTraits` для них пришлось бы четыре раза повторять один и тот же код. Решение проблемы предложено в данной библиотеке и основано на таких библиотеках, как `enable_if` и MPL (Metaprogramming Library) коллекции библиотек Boost [17].

Семейство шаблонов `enable_if` позволяет включать или исключать отдельные специализации шаблонов классов (а также шаблонов функций) на основе свойств аргументов шаблонов. Свойство шаблонного аргумента, которое необходимо проверять, — это принадлежность данного типа к множеству типов NTL, моделирующих конечные поля. Работа с наборами типов является одним из ключевых моментов метапрограммирования на шаблонах C++ и реализована в более старой библиотеке Loki (в виде «списков типов» — `type lists` из [15, гл. 3] — основополагающем труде по обсуждаемым в данной статье вопросам проектирования на C++ и в более современной — Boost. MPL (см. также `Sequence Classes` [4, 5.8]).

Вначале необходимо объявить коллекцию типов, принадлежность к которой будет проверяться далее:

```
typedef boost::mpl::vector<
    NTL::GF2E, NTL::ZZ_
    pE, NTL::GF2, NTL::ZZ_p>
    NtlFieldTypes;
```

Для использования `enable_if` в шаблон класса `CoefficientTraits` необходимо добавить один параметр, с помощью которого `enable_if` будет управлять специализациями `CoefficientTraits`:

```
template<typename CoefT, typename
    Enable = void>
struct CoefficientTraits {
    /* general implementation */
};
```

Затем остается предоставить специализацию, которая будет использоваться в случае, когда в качестве `CoefT` будет передан один из типов NTL:

```
template<typename T>
struct CoefficientTraits<
    CoefT,
    typename boost::enable_if<
        boost::mpl::contains<NtlFieldTypes, CoefT>
    >::type > {
    // specific implementation for NTL-
    types ...
};
```

2.3. Классы стратегий

Еще один прием, классы стратегий, обеспечивает параметризацию типа точки используемым в алгоритме мономиальным упорядочением, она основана на классах стратегий (policy classes) [15, гл. 1]. Стратегия — связанный набор действий, относящихся к разрабатываемому типу, ортогональный к остальной функциональности этого типа и нуждающийся в параметризации. Способ упорядочения точек N -мерной целочисленной решетки — хороший претендент на вынесение его кода в стратегию. Общая идея реализации состоит в том, чтобы добавить в разрабатываемый шаблонный тип еще один шаблонный параметр шаблона и наследоваться от него.

```
typename T, template <typename>
class Policy = DefaultPolicy> class
MyClass: public Policy<T> { /*...
*/};
```

В классической работе [15] предлагается публичное наследование, при котором составляющие базового класса попадают в интерфейс производного. В нашей реализации клиенту не предоставляется непосредственный доступ к функциям базового класса, вместо этого их вызов помещен в интерфейсные функции производного класса (в данном случае, типа точки). Например, преобразование данной точки в следующую за ней относительно мономиального упорядочения:

```
inline
Point<Dim, OrderPolicy>&
Point<Dim,
OrderPolicy>::operator++() {
    inc(data);
    return *this;
}
```

Функция `inc ()` определена в шаблоне `OrderPolicy`, от которого наследуется `Point` аналогично:

```
return totalLess (data, other. data);
```

Функция `totalLess ()` также определена в шаблоне `OrderPolicy`. Таким образом, шаблону `OrderPolicy` следует знать о типе поля `data` точки, который исполь-

зуется в качестве параметра шаблона `OrderPolicy`.

```
template<
    int Dim, // point dimension
    template <typename PointImpl>
class OrderPolicy =

    GradedAntilexMonomialOrder>
class Point : OrderPolicy<
    std::array<int, Dim> >
{ /* ... */};
```

2.4. C++11 и «функциональный» стиль STL

Наиболее интересные задачи, связанные с проектированием, возникают при реализации относительно низкоуровневых сущностей, таких как точки целочисленной решетки и полиномы. Реализация более высокоуровневых частей, связанных непосредственно с помехоустойчивым кодированием, более очевидна и выполняется в обычном стиле STL, который является одним из главных воплощений идей обобщенного программирования по Степанову. Стоит заметить, что с появлением в C++11 лямбда-функций программирование в стиле STL стало намного более удобным и прозрачным. Как указывает автор C++ Бьерн Страуструп в своем докладе: «STL и в целом обобщенное программирование многим обязаны функциональному программированию... (однако — авт.) функциональные объекты, (в дополнение к которым и пришли лямбда-функции C++11 — авт.), являются не самым элегантным способом выражения идеи функций высших порядков» [18, 4.1.4]. Приведем пример, демонстрирующий стиль STL и использование лямбда-функций в рассматриваемой библиотеке.

В ситуации декодирования одним из основных входных параметров является полученное по каналу «слово» r — n -мерный вектор над некоторым конечным полем. Синдром — это отображение, которое монотому m ставит в соответствие скалярное произведение r и вектора, полученного вычислением монома m в n точках P_i некото-

рой алгебраической кривой. С помощью вычисления синдрома в наборе мономов m_i , которые образуют базис в некотором специальном линейном пространстве, получается «синдромная последовательность»¹.

```
using namespace std::placeholders;
auto syndromComponentAtBasisElem =
    // declare lambda-function:
    [this,&received](BasisElem const &
be) ->
typename SyndromeType::value_type {
    auto tit =
        boost::make_
transform_iterator(
this->curvePoints.begin(),
        std::bind(
computeMonomAtPoint<Field,
BasisElem,
CurvePoint>,
        be,
        _1));
    return typename
SyndromeType::value_type(
        be,
        std::inner_
product(received.begin(), received.
end()),
tit, FieldElemTraits<Field>::addId
());
};

std::transform(basis.begin(),
basis.end(),
std::inserter(syn, syn.begin()),
syndromComponentAtBasisElem);
```

В первой строке импортируются имена из стандартного (начиная с C++11) пространства имен `std::placeholders`, которые позволяют использовать выражения вида `_1` при связывании (`std::bind`) отдельных аргументов функции (это еще одна черта функционального программирования в C++ — частичное применение функций). Далее начинается объявление лямбда-функции, ссылка на которую помещается в переменную `syndromComponentAtBasisElem`, автоматически определяемого компилятором

¹ Разрывы в операторных строках, не обусловленные форматом колонки, введены автором. — Прим. ред.

типа (`auto`). Лямбда-функция принимает на вход один аргумент типа `BasisElement`, который используется для обозначения мономов, и возвращает пару, состоящую из своего аргумента `be` и значения синдрома в этом мономе. Функция `boost::make_transform_iterator` создает специальный адаптер с интерфейсом итератора, который получается проходом переданного ей итератора по другой коллекции после применения ее переданной функции. В данном случае адаптируется коллекция точек кривой путем вычисления каждой точки в данном мономе `be` и в результате получается итератор по тому вектору, который будет участвовать в скалярном произведении, упомянутом в определении синдрома. Само скалярное произведение считается с помощью стандартного алгоритма `std::inner_product`. Синдромная последовательность вычисляется с помощью стандартного алгоритма `std::transform`.

В самом начале объявления лямбда-функции можно видеть перечисление параметров из контекста текущей функции, точнее, функции-члена класса `BMSDecoding`, которые она (лямбда-функции) захватывает: `[this, &received]`. Процесс такого захвата объектов объемлющего контекста носит в функциональном программировании название замыкания (`closure`), однако во многих функциональных языках перечисления захватываемых параметров не требуется. В языке C++ это является частью решения так называемой `upwards funarg problem` [19], связанной с тем, что захватывающая функция может иметь большую длительность жизни по сравнению с функцией, в которой были объявлены захватываемые параметры. Особо остро такая проблема стоит в языках, не обладающих средствами сборки мусора [20] (к каковым относится и C++), и не очень элегантное решение, связанное с явным перечислением аргументов, является, по всей видимости, неизбежной платой за использование механизма замыканий в C++.

Заключение

В статье продемонстрировано применение принципов обобщенного и метапрограммирования при проектировании и реализации алгоритмов из области вычислительной алгебраической геометрии (рекурсивное инстанцирование шаблонов, характеристики типов, классы стратегий, лямбда-функции C++11). Показано, как выразительные средства языка C++, реализующие указанные подходы, позволяют моделировать абстрактные математические понятия (например, многочлены многих переменных, точки многомерной дискретной решетки со специальным мономиальным упорядочением). Возможность практического решения задачи оптимизации программных систем, спроектированных на рассматриваемых принципах, представляет предмет отдельного исследования.

Список литературы

1. *Nøholdt T., van Lint J. H., Pellikaan R.* Algebraic geometry codes // Handbook of Coding Theory, V. S. Pless, W. C. Huffman, and R. A. Brualdi, Eds. Amsterdam, The Netherlands: Elsevier, 1998. P. 871–961.
2. *Madelung Y.* Implementation of a decoding algorithm for AG-codes from the Hermitian curve, Rept. IT 93–137, Inst. Circuit Theory Telecom., Tech. Univ. of Denmark, Lyngby, Denmark, Apr. 1993.
3. *Маевский А. Э., Пеленицын А. М.* О программной реализации алгебро-геометрического кода с применением алгоритма Сакаты. В сб. «Материалы X Международной научно-практической конференции «Информационная безопасность»», ч. 2. Таганрог: ЮФУ. 2008. С. 55–57.
4. *Abrahams D., Gurtovoy A.* C++ Template Metaprogramming: Concepts, Tools, and Techniques from Boost and Beyond. Addison-Wesley, 2004.
5. *Musser D. A., Stepanov A. A.* Generic Programming // Proceeding of International Symposium on Symbolic and Algebraic Computation. Vol. 358 of Lecture Notes in Computer Science. Rome. Italy. 1988. P. 13–25.
6. *Siek J. G., Lumsdaine A.* The Matrix Template Library: A generic programming approach to high performance numerical linear algebra // In International Symposium on Computing in Object-Oriented Parallel Environments, 1998.
7. *Siek J. G., Lee L.-Q., and Lumsdaine A.* The generic graph component library. In Proceedings of the 1999 ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications. ACM Press, 1999. P. 399–414.
8. *Veldhuizen T. L.* Arrays in Blitz++. // Proceedings of the 2nd International Scientific Computing in Object-Oriented Parallel Environments (ISCOPE'98), vol. 1505 of Lecture Notes in Computer Science. Springer-Verlag, 1998.
9. *Cox D. A., Little J. B., O'Shea D. B.* Using Algebraic Geometry, Second Edition. Springer, 2005. — 496 p.
10. *Пеленицын А. М.* О реализации n-мерного BMS-алгоритма средствами обобщенного программирования // Труды научной школы И. Б. Симоненко. Ростов н/Д: изд-во ЮФУ, 2010. С. 197–203.
11. International Standards Organization: Programming Languages — C++. International Standard ISO/IEC 14882:2011.
12. *Саттер Г., Александреску А.* Стандарты программирования на C++. М.: Вильямс, 2005. — 224 с.
13. *Бурбаки Н.* Алгебра (Многочлены и поля. Упорядоченные группы). М.: Наука, 1965. — 300 с.
14. *Вандевурд Д., Джосаттис Н.* Шаблоны C++: справочник разработчика. М.: Вильямс, 2003. — 544 с.
15. *Александреску А.* Современное проектирование на C++. М.: Вильямс, 2002. — 336 с.
16. *Shoup V.* NTL: A Library for doing Number Theory // URL: <http://shoup.net/ntl/>.
17. Boost C++ Libraries. URL: <http://www.boost.org/>.
18. *Stroustrup B.* Evolving a language in and for the real world: C++ 1991–2006 // Procs. of the ACM HOPL–III. 2007. P. 4–1–4–59.
19. *Moses J.* The Function of FUNCTION in LISP, or Why the FUNARG Problem Should be Called the Environment Problem // MIT AI Memo 199, 1970.
20. *Appel A. W., Shao Z.* An Empirical and Analytic Study of Stack vs. Heap Cost for Languages with Closures // Princeton CS Tech Report TR-450–94, 1994.

Астанин Сергей Васильевич — докт. техн. наук, профессор кафедры Информатики Таганрогского государственного педагогического института, astser@mail.ru

Власова Екатерина Аркадьевна — старший преподаватель кафедры Математических и инструментальных методов экономики Московского финансово-промышленного университета «Синергия», evlasova@mpra.ru

Емельянов Александр Анатольевич — докт. экон. наук, профессор, вице-президент, заведующий кафедрой Математических и инструментальных методов экономики Московского финансово-промышленного университета «Синергия», aemelianov@mpra.ru

Емельянова Наталия Захаровна — канд. экон. наук, доцент кафедры Экономики и управления Московского энергетического института (Технического университета), nemelianova@mpra.ru

Жуковская Наталья Константиновна — канд. техн. наук, заместитель директора Таганрогского филиала НОУ ВПО «Российский новый университет», nasha-0207@yandex.ru

Зимовец Ольга Анатольевна — аспирант кафедры Прикладной информатики Национального исследовательского университета «Белгородский государственный университет», OZimovets@bsu.edu.ru

Касимов Денис Рашидович — магистрант кафедры Автоматизированных систем обработки информации и управления Ижевского государственного технического университета, kasden@mail.ru

Кацман Валерий Евелиевич — докт. техн. наук, профессор, директор Учебно-методического центра подготовки профессиональных оценщиков Московского финансово-промышленного университета «Синергия», courses@mail.ru

Кожин Олег Владимирович — науч. сотрудник лаборатории Радиационного контроля Ведущего научно-исследовательского института химической технологии государственной корпорации «Росатом», аспирант кафедры Информационных технологий Московского государственного университета тонких химических технологий имени М. В. Ломоносова, lrkooos@gmail.com

Корнюшко Валерий Федорович — докт. техн. наук, профессор, заслуженный деятель науки РФ, заведующий кафедрой Информационных технологий Московского государственного университета тонких химических технологий имени М. В. Ломоносова, vfk256@mail.ru

Кузин Рудольф Евгеньевич — докт. техн. наук, профессор, начальник лаборатории Радиационного контроля Ведущего научно-исследовательского института химической технологии государственной корпорации «Росатом», профессор Московского государственного университета тонких химических технологий имени М. В. Ломоносова, rkuzin256@mail.ru

Кулыгина Марина Михайловна — канд. техн. наук, ведущий научный сотрудник, ведущий программист Информационно-вычислительного центра Московского государственного университета тонких химических технологий имени М. В. Ломоносова, mmk45@mail.ru

Кучуганов Александр Валерьевич — канд. техн. наук, доцент, докторант кафедры Автоматизированных систем обработки информации и управления Ижевского государственного технического университета, Aleks_KAV@udm.ru

Лебедев Игорь Витальевич — инженер лаборатории Радиационного контроля Ведущего научно-исследовательского института химической технологии государственной корпорации «Росатом», студент факультета автоматики и электроники Национального исследовательского ядерного университета «МИФИ», lrkooos@gmail.com

Любимов Евгений Валерьевич — канд. техн. наук, начальник отдела Информационных систем в управлении Департамента информационных технологий Дальневосточного федерального университета, г. Владивосток, Lubimov.ev@mail.ru

Маторин Сергей Игоревич — докт. техн. наук, профессор кафедры Прикладной информатики Национального исследовательского университета «Белгородский государственный университет», matorin@bsu.edu.ru

Митяков Евгений Сергеевич — аспирант кафедры Экономической информатики Нижегородского государственного университета им. Н. И. Лобачевского, snmit@mail.ru

Могирев Александр Максимович — науч. сотрудник Лаборатории радиационного контроля Ведущего научно-исследовательского института химической технологии государственной корпорации «Росатом»,

аспирант кафедры Автоматизированных систем управления Московского горного государственного университета, lrkooos@gmail.com

Норица Вера Михайловна — ведущий программист Информационно-вычислительного центра, аспирант естественнонаучного факультета Московского государственного университета тонких химических технологий имени М. В. Ломоносова, noritsav@rambler.ru

Озерова Галина Павловна — канд. техн. наук, доцент кафедры Механики и математического моделирования Дальневосточного федерального университета, г. Владивосток, Gal_o@bk.ru

Пеленицын Артём Михайлович — аспирант факультета математики, механики и компьютерных наук Южного федерального университета, г. Ростов-на-Дону, apel@sfedu.ru

Писаненко Сергей Сергеевич — инженер Лаборатории радиационного контроля Ведущего научно-исследовательского института химической технологии государственной корпорации «Росатом», студент естественнонаучного факультета Московского государственного университета тонких химических технологий имени М. В. Ломоносова, lrkooos@gmail.com

Попов Алексей Анатольевич — канд. техн. наук, доцент, заведующий кафедрой Информационных систем в экономике и менеджменте Российского экономического университета имени Г. В. Плеханова, г. Москва, a1710p@mail.ru

Прокимов Николай Николаевич — канд. техн. наук, доцент кафедры Математических и инструментальных методов экономики Московского финансово-промышленного университета «Синергия», prokimovnn@mail.ru

Разумовский Алексей Игоревич — канд. техн. наук, старший научный сотрудник лаборатории Компьютерной графики, сотрудник Института проблем управления РАН, г. Москва, razumowsky@yandex.ru

Сазонтов Владимир Александрович — канд. физ.-мат. наук, доцент кафедры Прикладной информатики Нижегородского государственного технического университета им. П. Е. Алексеева, snmit@mail.ru

Салтан Андрей Анатольевич — ассистент кафедры Информационных систем в экономике экономического факультета Санкт-Петербургского государственного университета, a.saltan@econ.spb.ru

Слива Максим Владимирович — канд. пед. наук, доцент кафедры Информатики и методики преподавания информатики Нижневартковского государственного гуманитарного университета, persimon@inbox.ru

Стефанова Наталья Александровна — аспирант, старший преподаватель кафедры Электронной коммерции Поволжского государственного университета телекоммуникаций и информатики, г. Самара, stefna@yandex.ru

Таиров Таир Надирович — канд. техн. наук, директор Санкт-Петербургского филиала центрального института повышения квалификации, докторант Московского института тонких химических технологий им. М. В. Ломоносова, Tair.Tairov@atomprof.spb.ru

Телемтаев Марат Махметович — докт. техн. наук, профессор кафедры Информационных систем в экономике и менеджменте Российского экономического университета имени Г. В. Плеханова, г. Москва, m.telemtaev@gmail.com

Трифонов Юрий Васильевич — докт. экон. наук, профессор, декан Экономического факультета, заведующий кафедрой Экономической информатики Нижегородского государственного университета им. Н. И. Лобачевского, decanat@ef.unn.ru

Филиппов Пётр Борисович — менеджер ЗАО «Лаборатория Касперского», аспирант кафедры Информационной безопасности Московского государственного института электроники и математики, filippov607@mail.ru

Храпов Игорь Викторович — канд. техн. наук, директор Аналитического центра экономического развития Тамбовского государственного технического университета, igor@admin.tstu.ru

Хубаев Георгий Николаевич — докт. экон. наук, профессор кафедры Экономической информатики и автоматизации управления Ростовского государственного экономического университета (РИНХ), gnh@donpac.ru

Astanin Sergey — Doctor of Engineering, Professor, Computer Science Department, State Teacher Training Institute, the city of Taganrog, astser@mail.ru

Emelianov Alexander — Doctor of Economics, Professor, Vice-president, Head of the Department of Mathematical and Instrumental Methods of Economics, Moscow University of Finance and Industry «Synergy», aemelianov@mfpa.ru

Emelianova Natalia — Ph. D. (Econ.), Associate Professor, Senior Lecturer, Department of Economy and Management, Moscow Power Engineering Institute (Technical University), nemelianova@mfpa.ru

Filippov Petr — Manager, CJSC «Kaspersky Lab», Moscow, Post-Graduate Student, Faculty of Applied Mathematics, Informational Security Department, Moscow Institute of Electronics and Mathematics, filippov607@mail.ru

Kasimov Denis — Master Student, CAD Systems Department, Izhevsk State Technical University, kasden@mail.ru

Katsman Valery — Doctor of Engineering, Professor, Director of Training Center of Professional Appraisers, Moscow University of Finance and Industry «Synergy», courses@mail.ru

Khrapov Igor — Ph. D. (Eng), Director of Analytical Center of Economic Development, Tambov State Technical University, igor@admin.tstu.ru

Khubaev Georgy — Doctor of Economics, Professor, Economic Informatics and Control Automation Department, Rostov-on-Don State Economic University RINH, gnh@donpac.ru

Kornyushko Valery — Doctor of Engineering, Professor, Honored Scientist of Russian Federation, Head of Information Technology Department, Moscow State University of Fine Chemical Technologies n. a. M. V. Lomonosov, vfk256@mail.ru

Kozhin Oleg — Master of Science, Research Fellow, Radiation Control Laboratory, Post-Graduate Student, Natural Science Faculty, Information Technologies Department, Moscow State University of Fine Chemical Technologies n.a. M. V. Lomonosov, lrkooos@gmail.com

Kuchuganov Alexander — Ph. D. (Eng.), Associate Professor, CAD Systems Department, Izhevsk State Technical University, Aleks_KAV@udm.ru

Kulygina Marina — Ph. D. (Eng.), Senior Researcher, Lead Programmer, Information Computing Center, Moscow State University of Fine Chemical Technologies n. a., M. V. Lomonosov, mmk45@mail.ru

Kuzin Rudolf — Doctor of Engineering, Professor, Head of Radiation Control Laboratory, Lead Research Institute of Chemical Technology of State Corporation «Rosatom», Professor of Information Technologies Department, Moscow State University of Fine Chemical Technologies n. a. M. V. Lomonosov, rkuzin256@mail.ru

Lebedev Igor — Engineer, Radiation Control Laboratory, Lead Research Institute of Chemical Technology of State Corporation «Rosatom», Student, Faculty of Automatics and Electronics, Micro- and Nanoelectronics Department, National Research Nuclear University MEPhI, Moscow, lrkooos@gmail.com

Lyubimov Evgeny — Ph. D. (Eng.), Head of Management Information Systems Department, Far Eastern Federal University, the city of Vladivostok, Lubimov.ev@mail.ru

Matorin Sergey — Doctor of Engineering, Professor of Applied Computer Science Department, Belgorod State University, matorin@bsu.edu.ru

Mityakov Evgeny — Assistant, Applied Mathematics Department, Post-Graduate Student, Economic Faculty, Economic Computer Science Department, Nizhny Novgorod State Technical University n. a. R. E. Alekseev, snmit@mail.ru

Mogirev Alexander — Master of Science, Research Fellow, Radiation Control Laboratory, Lead Research Institute of Chemical Technology of State Corporation «Rosatom», Post-Graduate Student, Automation and Information Science Faculty, Industrial Control System Department, Moscow State Mining University, lrkooos@gmail.com

Noritsa Vera — Lead Programmer, Information Computing Center, Post-Graduate Student, Natural Science Faculty, Information Technologies Department, Moscow State University of Fine Chemical Technologies n. a. M. V. Lomonosov, noritsav@rambler.ru

Ozerova Galina — Ph. D. (Eng.), Assistant Professor, Mechanics and Mathematical Modeling Department, Far Eastern Federal University, the city of Vladivostok, Gal_o@bk.ru

Pelenitsyn Artem — Post-Graduate Student, Faculty of Mathematics, Mechanics and Computer Sciences Department, Southern Federal University, the city of Rostov-on-Don, apel@sfedu.ru

Pisanenko Sergey — Engineer, Radiation Control Laboratory, Lead Research Institute of Chemical Technology of State Corporation «Rosatom», Student, Natural Science Faculty, Information Technologies Department, Moscow State University of Fine Chemical Technologies n. a. M. V. Lomonosov, Irkooos@gmail.com

Popov Alexey — Ph. D. (Eng.), Associate Professor, Head of Information Systems in Economics and Management Department, Plekhanov Russian Economic University, Moscow, a1710p@mail.ru

Prokimnov Nikolay — Ph. D. (Eng.), Associate Professor, Senior Lecturer, Department of Mathematical and Instrumental Methods of Economics, Moscow University of Finance and Industry «Synergy», prokimnovnn@mail.ru

Razumowsky Alexey — Ph. D. (Eng.), Senior Researcher, Computer Graphics Laboratory, Institute of Control Sciences (Russian Academy of Science), Moscow, razumowsky@yandex.ru

Saltan Andrey — Assistant, Economics, Information Systems Department, Faculty of Economy, St. Petersburg State University, a.saltan@econ.pu.ru

Sazontov Vladimir — Ph. D. (Math.), Associate Professor, Applied Mathematics Department, Nizhniy Novgorod State Technical University n. a. R. E. Alekseev, snmit@mail.ru

Sliva Maxim — Ph. D. (Ped.), Associate Professor, Informatics and it's Methods of Teaching Department, Nizhnevartovsk State University of Humanities, the city of Nizhnevartovsk, persimon@inbox.ru

Stefanova Natalia — Senior Lecturer, Post-Graduate Student, E-commerce Department, Povolzhskiy State University of Telecommunication and Information, the city of Samara, stefna@yandex.ru

Tairov Tair — Ph. D. (Eng.), Associate Professor, Director of St. Petersburg branch of Central Institute for Continuing Education and Training, Doctoral Candidate of Moscow State University of Fine Chemical Technologies n. a. M. V. Lomonosov, Tair.Tairov@atomprof.spb.ru

Telemtaev Marat — Doctor of Engineering, Professor, Information Systems in Economics and Management Department, Plekhanov Russian Economic University, Moscow, m.telemtaev@gmail.com

Trifonov Yuri — Doctor of Economics, Professor, Dean of Economic Faculty, Head of Economic Computer Science Department, National Research University «Nizhniy Novgorod state university of N. I. Lobachevsky», decanat@ef.unn.ru

Vlasova Ekaterina — Senior Lecturer, Department of Mathematical and Instrumental Methods of Economics, Moscow University of Finance and Industry «Synergy», evlasova@mfp.ru

Zhukovskaya Natalja — Ph. D. (Eng.), Deputy Director, Taganrog branch of Russian New University, nasha-0207@yandex.ru

Zimovets Olga — Post-Graduate Student, Applied Computer Science Department, Belgorod State University, OZimovets@bsu.edu.ru

Фрактальный анализ поведения быстрых индикаторов экономики России

Ю. В. Трифонов, Е. С. Митяков, В. А. Сазонтов

Статья посвящена статистическому анализу и прогнозу поведения быстрых индикаторов экономики России на основе методов фрактального анализа. Индикаторы образуют пять множеств: сводные показатели, финансы, цены, социальная сфера, производство важнейших видов промышленной продукции. Показано, что для анализируемых индикаторов характерна средняя длина цикла 6–7 лет. Даны прогнозы динамики индикаторов, основанные на расчете показателя Херста.

Информационное пространство, фрактальный анализ, показатель Херста, прогнозирование социально-экономической динамики.

Fractal analysis of Russian economy fast indicators behavior

Yu. Trifonov, E. Mityakov, V. Sazontov

The article deals with the Russian economy rapid indicators statistical analysis and their behavior forecast based on fractal analysis approach. Five sets of indicators were considered which comprise aggregates, finance, prices, social sector and major industrial goods production. It is shown that the length of six or seven years features the analyzed indicators average cycle. Indicators projections given are based on the Hurst index calculations.

Information space, fractal analysis, Hurst index prediction of socio-economic dynamics.

Моделирование рынка программного обеспечения при наличии внешнего сетевого эффекта и компьютерного пиратства

А. А. Салтан

В работе проанализированы основные характеристики рынка программного обеспечения и ценовой стратегии компаний-производителей программного обеспечения. На основе анализа построена модель, в динамике описывающая взаимодействие производителей и пользователей программного обеспечения при наличии рынка пиратских копий и внешнего сетевого эффекта. Модель позволяет компании-производителю программного обеспечения определить оптимальную ценовую политику.

Моделирование рынка, рынок программного обеспечения, пиратство программного обеспечения, внешние сетевые эффекты.

Software market modeling in the network externalities and digital piracy environment

A. Saltan

The software market and software manufacturing companies pricing strategies features are considered. A model describing the software producers and users interaction dynamics in the market assuming existence of pirated copies and external network effect is developed. The model may be used by software manufacturers as a tool of establishing the optimal pricing policy.

Market modeling, software market, software piracy, network externalities.

Совершенствование отечественных информационных систем управления недвижимостью на основе зарубежного опыта

М. А. Телемтаев, А. А. Попов

В статье рассмотрены функциональные возможности зарубежных информационных систем по управлению недвижимостью. Отмечены преимущества зарубежных систем по сравнению с аналогичными отечественными информационными системами. Обсуждаются целостные подходы к построению отечественных систем управления недвижимостью, не уступающих по функциональным возможностям зарубежным аналогам.

Информационная система, управление недвижимостью, целостный подход, абонентские веб-сервисы, удаленный доступ к сервисам.

Using international experience for improving domestic information systems of property management

M. Telemtaev, A. Popov

Foreign property management information systems features are considered. The advantages of these systems as compared with those of national information systems are discussed. Holistic approaches to building national information systems for real estate management which are not inferior to foreign analogues functionally are considered.

Information systems, property management, holistic approach, subscriber web-based applications, remote access to services.

Информационная поддержка системы радиационного контроля при большом потоке исследуемых проб

Р. Е. Кузин, О. В. Кожин, И. В. Лебедев,
А. М. Могирев, С. С. Писаненко, Т. Н. Таиров

Статья посвящена актуальным вопросам повышения уровня автоматизации работы лаборатории

радиационного контроля Ведущего научно-исследовательского института химической технологии. Разработанная авторами информационная система позволяет значительно сократить время обработки результатов и уменьшить фактор человеческой ошибки.

Контроль радионуклидного состава, информационная система, локальная вычислительная сеть.

Information support of radiation control system at a high flow of samples

R. Kuzin, O. Kozhin, I. Lebedev, A. Mogirev, S. Pisanenko, T. Tairov

The article deals with current tasks of increasing the laboratory of radiation control automation level in the Lead Research Institute of Chemical Technology. The information system developed by the authors can significantly reduce the time needed to process the laboratory output data as well as the number of human errors.

Control of radionuclide composition, information system, local area network.

Информационная поддержка мониторинга государственных закупок

В. Ф. Корнюшко, М. М. Кулыгина, В. М. Норица, И. В. Храпов

Предложена обобщенная системная модель, описывающая многоуровневую иерархическую структуру мониторинга государственных закупок. На ее основе разработана информационно-аналитическая база автоматизированной системы мониторинга с использованием технологии хранилищ данных и ориентированная на многомерный анализ данных.

Системная модель, хранилище данных, мониторинг, государственные закупки.

Information support for public procurement monitoring

V. Korniyushko, M. Kulygina, V. Noritsa, I. Khrapov

The unified model describing a multilevel hierarchical structure of the government procurement monitoring is proposed. Based on this model the information and analytical framework of automated monitoring system using data warehouse technology and focusing on multivariate data analysis is developed.

System model, data storage, monitoring, public procurement.

Кроссплатформенный подход как средство унификации обучения программированию в различных операционных системах

М. В. Слива

В статье представлен анализ кроссплатформенных технологий программирования (Java, Qt, GTK+)

и рассмотрены их достоинства и недостатки. Предложена методика поэтапного обучения с использованием кроссплатформенного подхода. Рассмотрена особенность перехода от структурного программирования к объектно-ориентированному с использованием библиотеки GTK+.

Кроссплатформенное программирование, Java, Qt, GTK+, методика обучения программированию.

Cross-platform approach as a means of teaching programming unification in a context of various operational systems

M. Sliva

The article concerns the cross-platform programming technologies such as Java, Qt, GTK+. Their advantages and disadvantages analyzed. The technique of gradual learning using cross-platform approach is presented. The main points of moving from structured programming to object-oriented programming using GTK+ library are described.

Cross-platform programming, Java, Qt, GTK+, methods of programming teaching.

Системный подход к управлению информатизацией университета

Е. В. Любимов, Г. П. Озерова,

В статье обсуждаются вопросы построения эффективной системы управления информатизацией университета, учитывающей его специфику и масштабы. Предлагается расширить известные сервис-ориентированные подходы управления информационными технологиями дополнительными механизмами и моделями, систематизирующими задачи информатизации и повышающими ее эффект.

Образование, информатизация, информационные технологии.

Using system approach to the University computerization management

E. Lyubimov, G. Ozerova

The issues of elaborating an efficient management system for University computerizing are discussed. Introducing additional mechanisms and models is proposed as an extension of well known information technology service-oriented approaches which provides the information problems set organizing and increases the final effects.

Education, informatization and information technology.

Оценка эффективности поисковой оптимизации сайтов вузов с использованием поисковых запросов

Н. А. Стефанова

Работа посвящена разработке методики оценки результатов поисковой оптимизации сайтов вузов

для двух типов запросов: тематических (рынок высшего образования) и по имени вуза. Производится анализ сайтов вузов самарского региона.

Вуз, сайт, поисковая оптимизация, индексы поискового запроса.

Indices of Search Queries

N. Stefanova

The research deals with the development of methodology that helps to rate the results of search engine optimization. The results are defined by two types of search queries: according to the field of search (institutions of high education) and according to the name of an institution. In the work the sites of higher education institutions in Samara region are analyzed.

High education, of search engine optimization, indices of search queries.

Методы обобщенного и метапрограммирования в программной реализации декодера алгебро-геометрических кодов

A. M. Пеленицын

Описываются проектные решения, принятые при разработке программной реализации декодера одного класса алгебро-геометрических кодов. Данные решения развивают методологию обобщенного программирования, а также демонстрируют ряд приемов метапрограммирования на языке C++, которые оказываются полезными при решении аналогичных задач из области помехоустойчивого кодирования или более общих задач компьютерной алгебры. Отмечаются инструменты C++11, призванные улучшить поддержку обобщенного программирования на языке C++.

C++, шаблоны, метапрограммирование, обобщенное программирование.

Generic and meta- programming methods in software implementation of decoder of algebraic geometry codes

A. Pelenitsyn

We describe design decisions adopted in our software implementation of decoder for a class of algebraic geometry codes. The solutions develop a methodology of generic programming and show some metaprogramming techniques which are valuable when solving similar problems from the field of error-correcting codes or more generally computational algebraic geometry. We also emphasize features of C++11 which are to improve support for generic programming in C++ language.

C++, templates, metaprogramming, generic programming.

Использование и защита персональных данных в социальных сетях Интернета

П. Б. Филиппов

В статье рассмотрены вопросы информационного обмена пользователей сети Интернет. Перечислены основные особенности взаимодействия субъектов социальных сетей и возникающие угрозы. Приведены статистические данные относительно роста числа пользователей Интернета и социальных сетей. Предложены шаги по созданию программного обеспечения, позволяющего пользователям социальных сетей предотвратить доступ к их персональным данным со стороны злоумышленников. Отмечены достоинства и недостатки реализованных и предлагаемых средств защиты от утечки персональных данных. Сделаны выводы о практической применимости результатов исследования.

Персональные данные, социальные сети, информационная безопасность, программное обеспечение.

Using and protecting of personal data in Internet social networks

P. Filippov

The paper deals with Internet users' communication. The main issues concerning social networks members and emerging communication threats are identified. Statistics on the growing number of Internet and social networks users is given. Steps to create software that allows users of social networks to prevent hackers' access to their personal information are proposed. The advantages and disadvantages of implemented and proposed tools for preventing personal data leakage are noted. The conclusions about the study results implementation are made.

Personal data, social networks, information security, software.

Практика создания 3D-тренажеров

A. И. Разумовский

В работе рассматриваются назначение и эффекты применения компьютерных 3D-тренажеров, средства и особенности их создания на основе средств виртуальной реальности. Приведены примеры программных тренажерных систем, разработанных при участии автора, представлены особенности их проектирования, показана практическая значимость применения 3D-тренажеров в широком спектре процессов промышленного производства и обучения

3D-тренажер, виртуальная реальность, компьютерная графика, обучение, моделирование движения, эргономика, геоинформационные системы, прочностные расчеты, 3D-визуализация.

The practice of creating 3D simulators

A. Razumowsky

This paper describes the purpose and benefits of 3D computer simulators, tools and features of their creation by means of virtual reality. Examples of the software simulator systems developed with the participation of the author present the features of their design. The practical importance of the use of 3D simulators in a wide range of industrial and education is shown.

3D simulator, virtual reality, computer graphics, learning, simulation of the motion, ergonomics, information systems, strength calculations, 3D-visualization.

Графический поиск чертежей в хранилищах данных

А. В. Кучуганов, Д. Р. Касимов

В работе представлен подход к решению задачи поиска чертежей (эскизов) на основе многоуровневого распознавания и логического анализа, использующий построение и сопоставление нечетких атрибутивных графов образов объектов. Подход базируется на методах и инструментарии структурного распознавания образов, нечетких графов, автоматического чтения чертежей и синтеза 3D-моделей, а также использовании лучевого графа в процессе распознавания. Приведены примеры.

Чертеж, распознавание образов, нечеткий граф, лучевой граф.

Graphical search of technical drawings over data warehouses

A. Kuchuganov, D. Kasimov

An approach to drawings (sketches) searching based on a multilevel recognition and logical analysis is presented. The approach incorporates construction and comparison of objects patterns fuzzy attribute graphs and relies on the methods and tools of structural pattern recognition, fuzzy graphs, automatic reading of drawings and 3D models as well as exploring beam graph in the recognition process. Examples are given.

Technical drawing, pattern recognition, fuzzy graph, beam graph.

Имитационное моделирование инвестиционных процессов

А. А. Емельянов, Е. А. Власова, Н. З. Емельянова, Н. Н. Прокимов,

Предложен комплексный подход к анализу инвестиционных процессов. Анализируется динамика взаимного влияния друг на друга инвестора и объекта инвестирования. Поддержку принятия инвестиционных решений обеспечивает система взаи-

модействующих имитационной и аналитической моделей, которые вместе с инструментальными методами образуют виртуальный моделирующий стенд. В качестве примера рассматривается инвестиционный процесс, связанный с муниципальным землепользованием.

Инвестиции, аналитическая модель, имитационная модель.

Simulating investment processes

A. Emelianov, E. Vlasova, N. Emelianova, N. Prokimnov

A method for investment processes analysis based on system approach is discussed. We analyze the dynamics of mutual investor and investment object influences. A decision-making supporting tool is composed of collaborative simulation and analytical models which in conjunction with application methods and techniques form the virtual modeling workbench. To illustrate the method an investment process associated with the municipal land use is considered.

Investment, analytical model, simulation model.

Формализованное визуальное моделирование административных процедур

С. И. Маторин, О. А. Зимовец

В статье рассмотрены оригинальные способы системного графоаналитического и алгебраического описания административных процедур с целью обеспечения оказания государственных и муниципальных услуг населению в электронном виде в рамках выполнения программы «Электронная Россия». Представлен метод формального описания систем в терминах «Узел», «Функция», «Объект».

Системный анализ, визуальное графоаналитическое моделирование, административные процедуры, подход «Узел — Функция — Объект», теория паттернов, исчисление процессов.

Formalized visual modeling of administrative procedures

I. Matorin, O. Zimovets

New methods for graph-analytical and algebraic description of administrative procedures are considered. The administrative procedures being analyzed provide public and municipal services in electronic form within the framework of «Electronic Russia» federal program. A method for systems formal description in terms of «unit», «function» and «object» is presented.

System analysis, visual graph-analytical modeling, administrative procedures, approach «Unit — Function — Object», patterns theory, calculation process.

Оценка резервов снижения ресурсоемкости товаров и услуг: методы и инструментальные средства

Г. Н. Хубаев

Предложен метод оценки резервов снижения ресурсоемкости товаров и услуг, требующий минимальных затрат. Показано, что имитационное моделирование позволяет получать статистические характеристики и распределение затрат ресурсов при любом распределении ресурсоемкости отдельных операций процесса. Предлагается оценивать резервы снижения ресурсоемкости товаров и услуг по величине коэффициента вариации, величине правосторонней асимметрии и медианы эмпирического распределения затрат ресурса.

Резервы снижения ресурсоемкости товаров и услуг, имитационное моделирование, величина правосторонней асимметрии и медианы распределения, процессно-статистический учет затрат ресурсов.

Evaluating the reserves to reduce resource consumption for producing goods and services: methods and tools

G. Khubaev

A method for estimating reserves reducing the resource consumption of goods and services which requires minimal resources is suggested. It is shown that the simulation may provide obtaining both statistical characteristics and the overhead distribution of each resource-allocation process. It is proposed to estimate reserves to reduce resource consumption of goods and services on the magnitude of the coefficient of variation, the value of right-sided asymmetry and the median of the empirical distribution of the resource costs.

Reserves of goods and services resource consumption decreasing, simulation, distribution right-side skewness and distribution median, process-statistical calculation of resources consumption.

Внутрифирменные механизмы распределения ограниченных ресурсов на основе переговорного процесса

С. В. Астанин, Н. К. Жуковская

В работе рассматриваются механизмы распределения ограниченных ресурсов в иерархической системе управления. Предложена обобщенная модель поведения элементов иерархической системы, состоящая из двух частей. Первая часть ба-

зируется на понятии нечеткой компромиссной игры, позволяющей n исполнителям строить общую стратегию распределения m ресурсов между собой на основе переговоров. Во второй модели поставлена задача согласования интересов центра и исполнителей на основе равновесных по Нэшу ситуаций.

Организационная система, распределение ресурсов, согласование интересов, нечеткая игра, равновесные стратегии, компромисс.

Tools of enterprise finite resources distribution on the basis of negotiation

S. Astanin, N. Zhukovskaya

Finite resources distribution mechanisms in a hierarchical control system are considered. The generalized model of system elements behavior consisting of two parts is suggested. The first model incorporates the concept of the fuzzy compromise game allowing performers to build the general resources distribution strategy using negotiations. The second model aims to harmonizing the interests of the center and performers on the basis of Nash balanced situations.

Organizational system, distribution of resources, coordination of interests, fuzzy game, equilibrium strategies, compromise.

Модель формирования оптимальной структуры аппарата управления организации

В. Е. Кацман

Организация анализируется как система, которая моделируется с применением теории графов. Рассмотрены граф управления и граф информации. Представлен алгоритм построения оптимальной структуры управленческого аппарата организации.

Организация, система, теория графов, граф управления, граф организации, аппарат управления организации.

The model of constructing the optimal organization management structure

V. Katsman

An enterprise management structure is modeled using graph theory approach. The management graph and the information graph are considered. An algorithm for constructing the optimal organization management structure is presented.

Organization, system, graph theory, management graph, information graph, organization management.

Правила оформления рукописей для представления в редакцию журнала «Прикладная информатика»

В связи с требованиями Высшей аттестационной комиссии (ВАК) Минобрнауки России в отношении рецензируемых периодических изданий редакция публикует правила представления материалов в журнал «Прикладная информатика». Правила введены в действие с 1 июля 2010 года для всех поступающих статей.

В журнал «Прикладная информатика» представляются статьи, содержащие новые научные и практические результаты по разделам:

- корпоративные информационные системы и технологии;
- разработка новых приложений, внедрение информационных систем и технологий в различных отраслях экономики;
- информационные и коммуникационные среды бизнеса, электронная коммерция;
- программные средства и программная инженерия;
- компьютерные методы и технологии электронного образования: e-education, e-learning;
- виртуальная реальность: профессиональные тренажеры и компьютерные игры;
- компьютерное моделирование процессов: modeling и simulation;
- проблемы информационной безопасности;
- математические и инструментальные методы экономики (при условии обязательной привязки к информатике);
- искусственный интеллект и обработка знаний;
- теория систем и системный анализ;
- новые методы и инструментальные средства информатики;
- репортажи, очерки, хроника, выставки, письма в редакцию, дискуссии, новые книги.

Редакционный совет журнала заинтересован в опубликовании статей научного и практического характера, в которых представлены новые результаты или разработки в области IT, информационных систем, баз данных или software в экономике, менеджменте, e-commerce. Таким статьям отводится до 75% объема журнала.

Формулируя Правила, редакция предполагает, что авторы, специализирующиеся в прикладной информатике, владеют необходимыми навыками работы с современными IT-пакетами, рекомендуемыми для оформления рукописей.

Надеемся, что авторы воспримут наши правила и станут следовать им на практике. Статьи, подготовленные без соблюдения редакционных требований, будут рассматриваться в последнюю очередь, т. е. продолжительное время находиться в редакционном портфеле, теряя свою актуальность.

Условия опубликования статьи

1. Научно-практические статьи, представляемые в «Прикладную информатику», независимо от их объема **публикуются бесплатно**. Статьи рекламного содержания, рекламные модули или вставки помещаются в журнал на платной основе согласно утвержденным расценкам или на компенсационных условиях. Общий объем рекламы в одном номере журнала не должен превышать 5 полос.

2. Статья должна соответствовать данным Правилам.

3. Материалы публикуются только после положительной рецензии. Рецензент назначается главным редактором или его заместителем. Отрицательная рецензия может

быть предоставлена автору. Рецензирование является для автора анонимным.

4. Сроки опубликования статей зависят от величины очереди, которая образуется в связи с интенсивностью поступления статей в редакцию с учетом тематической компоновки редакционного портфеля.

Статьи предоставляются в электронном виде на e-mail редакции.

Структура рукописи

1. Статья должна начинаться с вводной части (введения), которая включает в себя содержательную постановку рассматриваемого вопроса, краткие сведения из его истории, разъяснения относительно того, где и когда изучаемый вопрос возникает. Должен быть ясен мотив, побудивший автора написать статью.

2. В основной части текста дается подробная постановка задачи, в том числе с позиций прикладной информатики. Если вопрос сводится к анализу некоторой модели, то должно быть пояснено, как эта модель вытекает из содержательной постановки задачи. Приводимые утверждения и результаты должны быть изложены и обстоятельно разъяснены.

При написании статьи следует придерживаться специальной терминологии, характерной для той области знаний, тематике которой посвящена статья.

Используемые в основном тексте редко встречающиеся специальные термины и обозначения необходимо разъяснять.

Не рекомендуется чрезмерное употребление аббревиатур, кроме общепринятых (за исключением тех редких случаев, когда автор претендует на открытие нового научного направления). Все аббревиатуры должны быть расшифрованы по мере их появления в тексте.

В статьях значительного объема рекомендуется использовать подзаголовки (2 уровня).

3. Заключительная часть статьи (заключение) должна содержать выводы, обсужде-

ние полученных результатов и, если возможно, пример, иллюстрирующий их эффективность, способы применения и практическую направленность.

4. Основной текст сопровождается рисунками (с подрисовочными подписями), формулами и таблицами, списком литературы.

Редакция не принимает к публикации заметки, тезисы и доклады от первого лица.

Материалы статьи формируются в текстовом редакторе MS Word (версий 6.0 и более поздних) и предоставляются в стандартном формате DOC или кросс-формате RTF. Формат листа А4, размеры полей: левого, правого, верхнего, нижнего — по 2 см. Шрифт Times New Roman размером 14 pt. Межстрочный интервал — полуторный (1,5). Нумерация страниц обязательна.

К статье прилагаются:

- заглавие на русском и английском языках;
- аннотация объемом 300–500 знаков по-русски и по-английски;
- сведения об авторах на русском и английском языках: ФИО, должность, наименование организации, почтовый адрес (включая индекс).

Таблицы

Названия строк и столбцов таблицы и ее заголовков должны быть краткими, но без сокращений. Таблицы анонсируются автором по тексту статьи.

Все данные, представленные в таблице, набираются в формате: одно значение — одна ячейка, одна строка (количество ячеек равно количеству столбцов).

Пример оформления таблицы:

Таблица 1

Бесплатные аналоги наиболее популярных приложений

| № | Назначение | Платные программы | Бесплатные аналоги |
|---|------------|-------------------|--------------------|
| 1 | | | |
| 2 | | | |
| 3 | | | |

Формулы

Создаются средствами встроенного в MS Word (до версии 2007) формульного редактора Equation или внешнего MathType с использованием стандартных настроек.

Формулы, набранные **во встроенном редакторе MS Word 2007**, в работу **не принимаются**.

Пример 1. Сложная, но правильно записанная формула с помощью Equation:

$$D_{B_{x \in G_p}} = D_q \delta T_x \left[\sum_{i=1}^N (k_{xi} H_i) + H_{xM} \right]. \quad (1)$$

Номера формул указываются справа в круглых скобках.

Недопустимо в отдельной строке создавать формулы или их части другими редакторами или с помощью печатных символов.

Не допускается создание формульных выражений с помощью составных символов в строке (или строках) или при помощи векторных редакторов.

Пример 2. Некорректная запись формульного выражения:

$$N = \sum_{i=1}^m \frac{k_i}{d_i}. \quad (1)$$

Пример 3. Корректная запись этого же формульного выражения:

$$N = \sum_{i=1}^m \frac{k_i}{d_i}. \quad (1)$$

Допускается набор специальных знаков и символов греческого алфавита при помощи системного символьного шрифта Symbol.

Стиль формул и переменных в тексте:

- цифры и греческие буквы, скобки в формулах, стандартные обозначения типов: sin, cos, log, e (основание натурального логарифма) пишутся прямо;
- латинские буквы (английский алфавит) набираются светлым курсивом;
- греческие буквы в формулах — прямым начертанием.

Рисунки

Иллюстративный материал желательнее представлять в виде объектов высокого разрешения. Громоздкие надписи на рисунке нужно размещать по тексту или в подрисовочных подписях. Не рекомендуется использовать графический редактор MS Word: из-за некачественной привязки текстов к деталям рисунка изображение искажается.

Максимальный размер рисунка (см): 15 (ширина) × 20 (высота). Если автор предполагает рисунок компактным, то по ширине он не должен превышать 7,2 см (ширина колонки).

Рекомендуемые графические редакторы:

- для создания векторных иллюстраций (блок-схем, графиков, рисунков) — пакеты Adobe Illustrator, Corel Draw. Допускается также выполнение схем средствами редакторов MS Word или Visio.
- для создания растровых иллюстраций и обработки отсканированных материалов — Adobe PhotoShop, Corel Photo-Paint; экранных форм (копий экрана) — любые программы захвата изображения (например, Corel Capture, который входит в состав пакета CorelDRAW Graphics Suite).

Векторные изображения предоставляются в следующем формате: толщина основных линий — 0,5 пункта (0,176 мм), шрифт надписей в элементах рисунка: Arial, размером 9 пт. Если иллюстрации представлены в стандартном графическом редакторе MS Word, они должны быть сгруппированы; если в формате внешнего редактора, — каждую векторную иллюстрацию нужно сохранить в отдельном файле и предоставить в исходном формате того графического средства, в котором иллюстрация была изначально выполнена. В названии файла следует отразить имя автора и порядковый номер рисунка (например, **Петров_Рис_1**).

Экранные формы должны быть подготовлены в соответствии с требованиями к разрешению растровых изображе-

ний. Прежде чем копировать изображения с экрана, следует установить максимальное разрешение экрана вашего монитора. Для этого через кнопку «Пуск» нужно выполнить: Панель управления→Экран→Параметры и задать самое большое разрешение экрана из допустимых на данном компьютере (рис. 1). Копирование окон в буфер обмена можно осуществлять с помощью предварительно загруженной программы Corel Capture (рис. 2).

При первой ее загрузке необходимо установить для изображений (Image) параметр разрешения (Resolution), равный 300 dpi, а также ширину (Width) и высоту (Height) изображения (в пикселях), равные максимальному разрешению вашего экрана (на рис. 1 это 1600 и 1200 точек соответственно). При загрузке Corel Capture можно установить режимы получения качественного изображения:

- 1) только текущее окно;
- 2) меню;
- 3) произвольный фрагмент экрана.

Формат изображения при первой загрузке Corel Capture устанавливается с параметрами: RGB (24 бит), точный размер, единицы — пиксели; ширина, высота — 100%. Такая настройка, как правило, выполняется только 1 раз. Перехват изображения далее производится клавишей F7. В результате автоматически получается растровая информация в буфере обмена с разрешением 300 dpi, которую нужно передать для последующей обработки или сохранения в виде файла в Corel Photo-Paint или Adobe PhotoShop в формате tif (без сжатия) или в формате jpg.

Для получения растрового изображения с помощью опции «PrtSc» следует открыть активное изображение, нажать сочетание клавиш «Shift+PrtSc», далее открыть редактор пиксельной графики, создать пустой файл с параметрами: разрешение — 300 dpi; цветовая модель 24 bit RGB, и вставить из буфера обмена данную копию, затем отредактировать изображение и сохранить его в формате tif (без сжатия) или jpg.

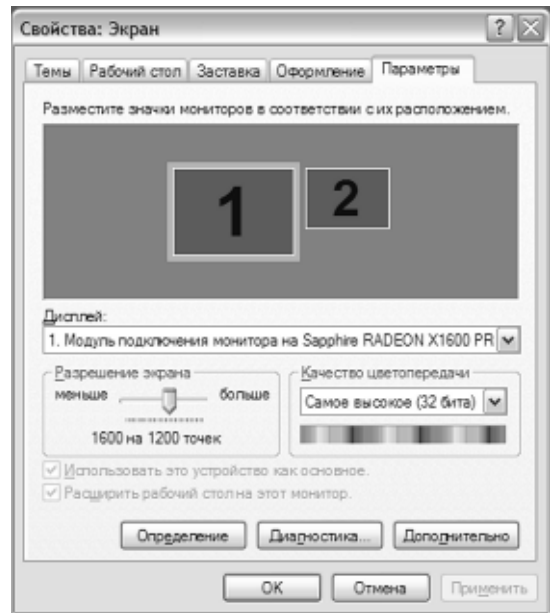


Рис. 1. Установка разрешения экрана



Рис. 2. Окно настроек Corel Capture

Снимки с цифровых фотоаппаратов прилагаются к тексту статьи как растровые jpg-файлы с разрешением не ниже 300 dpi.

Править (корректировать) фотографии следует с помощью Adobe PhotoShop, Corel

PhotoPaint или специального программного обеспечения, прилагаемого к фотоаппарату.

Снимки на фотобумаге должны быть отсканированы с разрешением не ниже 300 dpi.

Иллюстрации, заимствованные автором из других источников, должны иметь соответствующие ссылки.

Список литературы

Оформляется по принятому в журнале стандарту. Работы в библиографическом списке нумеруются по алфавиту, причем сначала перечисляются российские источники, а затем — иностранные либо в порядке следования ссылок. Номера ссылок в тексте заключаются в квадратные скобки.

Примеры оформления списка литературы:

а) книга:

Гиленсон П. Г. Справочник технического редактора. М.: Книга, 1972;

б) книга коллектива авторов:

Емельянов А. А., Власова Е. А., Дума Р. В. Имитационное моделирование экономических процессов / Под ред. А. А. Емельянова. М.: Финансы и статистика, 2004;

в) статья в книге типа «сборник трудов» или «сборник статей»:

Иванов А. А., Петров И. С. Электронная коммерция. В кн.: Современный бизнес. М.: МАКС Пресс, 2004;

г) статья в журнале:

Земляков С. Д., Рутковский В. Ю. Функциональная управляемость и настраиваемость систем координатно-параметрического управления // Автоматика и телемеханика. 1986. № 2;

д) доклад в сборнике трудов конференции:

Desai M., Ray A. A fault detection and isolation methodology // Proc. 20-th IEEE / Conf. On Decision and Control. San Diego, 1981.

Возможны также ссылки на электронные носители. Если материал представляет собой электронную публикацию (имеет заго-

ловок и авторов), он помещается в составе списка литературы с указанием ссылки на сайт-источник. Если же в статье используются какие-либо данные, предоставляемые электронным ресурсом, то предпочтительнее оформить ссылку на этот ресурс в виде концевой сноски по тексту статьи.

Доработка статьи

При необходимости доработки дата готовности статьи в редакции меняется, и, следовательно, отодвигается срок ее опубликования. Если автор согласен внести изменения в статью, то при представлении в редакцию нового варианта необходимо приложить и первоначальный вариант.

Принятие к публикации

В адрес автора направляется электронный экземпляр рукописи статьи с замечаниями научного редактора. Исправленный автором вариант возвращается на e-mail редакции и считается окончательным с содержательной точки зрения.

В случае значительных изменений автором отредактированного текста статья будет снята с рассмотрения на предмет ближайшей публикации и перенесена на последующую.

Замечания рецензентов

В случае отрицательной рецензии на рукопись статьи редколлегия вправе отказать автору в опубликовании этой статьи, а также имеет право оставить у себя электронный экземпляр рукописи (или один экземпляр статьи в бумажном варианте).

Уважаемые авторы!

Коллектив редакции надеется на вашу аккуратность в выполнении изложенных требований.

При возникновении вопросов или сомнений просим оперативно связаться с редакцией (в противном случае возможны потери времени на переделку, переписывание или перерисовку материалов статьи).

E-mail: appliedinform@marketds.ru.

Телефон: (495) 663-93-88 доб. 1833.

Подписка-2012

Журнал «Прикладная информатика» выходит 6 раз в год:
Февраль Апрель Июнь Август Октябрь Декабрь

Подписка через редакцию

Стоимость подписки на 2012 год:

| | | | | | |
|----------|-----------|-----------|-----------|-----------|-----------|
| 1 номер | 2 номера | 3 номера | 4 номера | 5 номеров | 6 номеров |
| 980 руб. | 1960 руб. | 2940 руб. | 3920 руб. | 4900 руб. | 5880 руб. |

Подписку можно оформить с любого месяца.

Тел./факс: (495) 663-93-88 (доб. 1839, 1845)

Руководитель службы маркетинга: *Н. М. Ларионова*

E-mail: nlarionova@mfpa.ru

Подписка на почте

По каталогу агентства «Роспечать» индекс 20497

По объединенному каталогу «Пресса России» индекс 88059

По каталогу российской прессы «Почта России» индекс 14241

Доставка осуществляется заказной бандеролью с уведомлением.

Электронный выпуск, а также отдельные статьи журнала можно приобрести на сайтах www.elibrary.ru и www.dilib.ru.

К оплате принимаются все виды электронных платежей, банковские карты.

Возможна также оплата с помощью SMS.

Учредитель и издатель ООО «Синергия ПРИНТ»
Свидетельство о регистрации ПИ № ФС77-48221 от 19.01.2012 г.

Литературный редактор *А. К. Наумко*

Верстка, дизайн макета *Б. В. Зипунов*

Адрес редакции

105318, г. Москва ул. Измайловский Вал д. 2 (юрид.)

125190, Москва, Ленинградский просп., д. 80, корп. Г, офис 612(4)

Тел.: (495) 663-93-88 (доб. 1833, 1839)

e-mail: appliedinform@marketds.ru; www.appliedinformatics.ru

Наши реквизиты

ООО «Синергия ПРИНТ»

ИНН 7702267103

КПП 771901001

ОГРН 1027700400375

Р/с 40702810000000012018

ОАО «Московский кредитный банк», г. Москва

К/с 30101810300000000659

БИК 044585659

При перепечатке и цитировании материалов ссылка на журнал «Прикладная информатика» обязательна.
Редакция не несет ответственности за достоверность информации, опубликованной в рекламных объявлениях.
Мнения авторов и редакции могут не совпадать.

© ООО «Синергия ПРИНТ»

Подписано в печать: 12.04.2012.

Тираж 3000 экз.

Отпечатано в ООО «Галлея-Принт».
111024, Москва, ул. 5-я Кабельная, д. 2Б.
Заказ № 351.