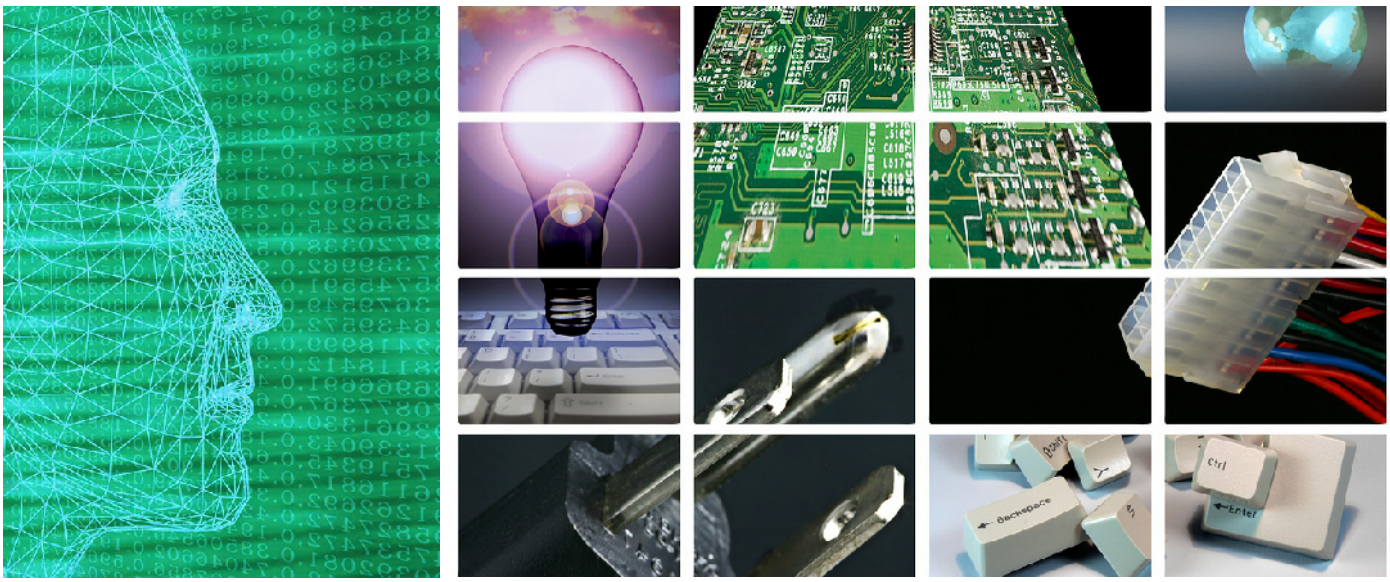
ТЕХНИЧЕСКИ УНИВЕРСИТЕТ - ВАРНА
TECHNICAL UNIVERSITY OF VARNA

Година XI, Брой 1/2013

КОМПЮТЪРНИ НАУКИ И ТЕХНОЛОГИИ



Bulgaria Communications Chapter

Faculty of Computing & Automation

COMPUTER SCIENCE AND TECHNOLOGIES

Юбилейна научна конференция с международно участие
45 години катедра "Компютърни науки и технологии"
30 години специалност "Компютърни системи и технологии"
27-28 септември, 2013 г. Варна, България

Year XI, Number 1/2013

Компютърни науки и технологии

Издание
на Факултета по изчислителна техника и автоматизация
Технически университет - Варна

Редактор: гл. ас. д-р Ю. Петкова
Гл. редактор: доц. д-р П. Антонов

Печат: ТУ-Варна

За контакти:

Технически университет - Варна
ФИТА
ул. „Студентска” 1, 9010 Варна,
България
тел./факс: (052) 383 320
e-mail: peter.antonov@ieee.org
jppet@abv.bg

ISSN 1312-3335

Computer Science and Technologies

Publication
of Computing and Automation Faculty
Technical University of Varna

Editor: Ass. Prof. Y. Petkova, PhD
Chief Editor: Assoc. Prof. P. Antonov, PhD

Printing: TU-Varna

For contacts:

Technical University of Varna
Faculty of Computing and Automation
1, Studentska Str., 9010 Varna,
Bulgaria
Tel/Fax: (+359) 52 383 320
e-mail: peter.antonov@ieee.org
jppet@abv.bg

ISSN 1312-3335

СБОРНИК ДОКЛАДИ

Юбилейна научна конференция с международно участие
45 години катедра "Компютърни науки и технологии"
30 години специалност "Компютърни системи и технологии"
27-28 септември, 2013 г.
Варна, България

Anniversary Scientific International Conference 45 Years Computer Sciences and Engineering Department
30 Years Computer Systems and Technologies Speciality
27-28 September, 2013
Varna, Bulgaria

PROCEEDINGS

Уважаеми колеги,

През настоящата 2013 година се навършват 45 години от създаването в нашия Университет на първото структурно звено “Изчислителна техника”, получило задачата да обучава студентите в областта на най-революционната научно-техническа област, тази на компютърните науки и технологии. Това звено извървя дълъг и възходящ път на развитие, за да достигне до настоящото си състояние и възможности в лицето на катедра “Компютърни науки и технологии”, чийто високо квалифициран преподавателски състав провежда съвременен и актуален по съдържание и форми учебен и научноизследователски процеси във всички образователно квалификационни степени - бакалавър, магистър и доктор. Катедрата притежава богата материална база и обучава студенти в две специалности, чийто общ брой надхвърля 500.

През тези години катедра КНТ завоюва безспорни и значими позиции в областта на компютърното образование и се утвърди като авторитетно звено сред аналогичните в страната. Всички успехи се дължат на всеотдайния и високо квалифициран труд на преподавателите и служителите на катедрения колектив. Неоспорима е заслугата и на нашите възпитаници - студенти и докторанти, защитили достойно престижа на катедрата и специалността както в страната, така във всички краища на света.

С уважение и благодарност се обръщам към основателите на катедрата. Не се съмнявам, че катедреният колектив ще съхрани и развие добрите традиции въпреки трудностите, тъй като приемствеността на младите в катедрата е налице.

С пожелание за здраве, щастие и успех във всички добри начинания!

Честита годишнина!

Септември, 2013 г.

доц. д-р инж. Надежда Рускова,
Ръководител катедра КНТ

Dear colleagues!

In the current 2013 45 years have passed since the first structure section on Computing at our University was established. It received the job to introduce the students in the most revolutionary scientific area – the one of computer science and technology. The Section has gone a long way up while reaching its present state and opportunities offered by the Department of Computer science and engineering, whose highly qualified lecturers hold a contemporary in content and shape academic and research process in all degrees - Bachelor, Master, PhD. The department has great facilities and teaches students, whose total number exceeds 500, in two subjects.

During these years the Department of “Computer science and engineering” gained indisputable importance in the field of computer education and established itself as an authoritative among similar across the country. All the success is due to the dedicated and highly skilled work of the lecturers and employees of the department. Undeniable is also the contribution of our undergraduate and graduate students, who have successfully defended the reputation of the Department and the subject both in the country and all over the world.

With respect and gratitude, I refer to the founders of the Department. I have no doubt that the Department staff will preserve and develop the traditions despite the difficulties, as the succession of the young people in the Department is available.

With wishes for health, happiness and success in all your good endeavors!

Happy anniversary!

September, 2013

Assoc. Prof. Nadezhda Ruskova, PhD
Head of Computer Science and Engineering Department

Организационен комитет	Organizing committee
Председател:	**Chairman:**
Надежда Рускова – ръководител на катедра „Компютърни науки и технологии”	**Nadezhda Ruskova** – Head of Computer Science and Engineering Department
Членове:	**Members:**
Слава Йорданова	Slava Yordanova
Христо Вълчанов	Hristo Valchanov
Виолета Божикова	Violeta Bozhikova
Христо Ненов	Hristo Nenov
Марияна Стоева	Mariana Stoeva
Венета Алексиева	Veneta Aleksieva
Емил Маринов	Emil Marinov
Ивайло Пенев	Ivaylo Penev
Юлка Петкова	Yulka Petkova
Стефка Попова	Stefka Popova
Милена Карова	Milena Karova
Сузан Четинова	Suzan Chetinova
Преслав Тодоров	Preslav Todorov

Програмен комитет	Programme committee
Председател:	**Chairman:**
Петър Антонов – декан на Факултета по изчислителна техника и автоматизация	**Peter Antonov** – Dean of Computing and Automation Faculty
Членове:	**Members:**
Елена Рачева	Elena Razcheva
Алексей Алексеев	Aleksey Alekseev
Димитър Тянев	Dimitar Tyanev
Михаил Шестопалов	Mikhail Shestopalov
Митко Митев	Mitko Mitev
Владимир Фетисов	Vladimir Fetisov
Трифон Русков	Trifon Ruskov
Ремир Сольницев	Remir Solnitsev
Анатолий Антонов	Anatoliy Antonov
Радек Кнофлицек	Radek Knoflicek
Петър Петров	Peter Petrov
Дамир Имаев	Damir Imaev
Сава Иванов	Sava Ivanov
Любомир Сотиров	Lyubomir Sotirov

СПОНСОРИ НА КОНФЕРЕНЦИЯТА

Telecoms http://www.telecoms.bg	**ТЕЛЕКОМС ООД** е телекомуникационен оператор, предоставящ услуги на регионални и локални интернет доставчици.
EuroRisk Systems Ltd. http://www.eurorisksystems.com	**EuroRisk Системи ООД** предоставя софтуерни системи и услуги за банки, застрахователни и финансови институции и други доставчици на финансови услуги.
ASIC DEPOT http://www.asicdepot.com	**ASIC Depot** предоставя разширени услуги по проектиране, дизайн и верификация на мрежови процесори, мултимедия, контролери и периферия, мобилни приложения, софтуер за проверка и др.
ADASTRA http://www.bg.adastragrp.com	**АДАСТРА** предлага информационен мениджмънт - пълна гама услуги, насочени към рационализация и ускоряване на бизнес процеси или обновяване на фирмена технологична инфраструктура.
SIS TECHNOLOGY AD BUSINESS & RETAIL SOLUTIONS http://www.sistechnology.com	**СИС Технология АД** предлага консултации, проучване, разработка, внедряване и поддръжка на системи за автоматизиране на търговски обекти, счетоводна отчетност, управление на човешки ресурси и проекти, изискващи висока степен на надеждност при обработката на операции в реално време.
CISCO Networking Academy http://ciscoacademy.tu-varna.bg	Локалната **Cisco Академия** при ТУ-Варна провежда курсове по най-новите учебни програми на Cisco Networking Academy - Cisco CCNA Exploration и Cisco CCNA Security на български и английски език.

СЪДЪРЖАНИЕ **CONTENTS**

СЕКЦИЯ 1
КОМПЮТЪРНИ СИСТЕМИ И МРЕЖИ

Юбилейна научна конференция с международно участие
45 години катедра "Компютърни науки и технологии"
30 години специалност "Компютърни системи и технологии"
27-28 септември, 2013 г.
Варна, България

Anniversary Scientific International Conference 45 Years Computer Sciences and Engineering Department
30 Years Computer Systems and Technologies Speciality
27-28 September, 2013
Varna, Bulgaria

SECTION 1
COMPUTER SYSTEMS AND NETWORKS

МИКРОПРОЦЕСОРНА СИСТЕМА ЗА ИЗМЕРВАНЕ ПАРАМЕТРИ НА МИКРОКЛИМАТА

Илия Ив. Хаджидимов, Ирина П. Павлова, Кръстин Кр. Йорданов

Резюме: В настоящата разработка е демонстрирана микропроцесорна система за регистриране и записване на основни параметри на микроклимата в помещения. Параметрите, които се измерват са температурата на въздуха, плътността на потока глобална слънчева радиация, относителното влагосъдържание на въздуха и осветеността на помещението. Микропроцесорната система може да работи самостоятелно без персонален компютър, а данните се съхраняват на магнитен носител. С помощта на тази система могат да се оценяват параметрите на микроклимата в помещения с различно предназначение и резултатите да се сравняват с нормативните изисквания.
Ключови думи: Параметри на микроклимат, микропроцесорна система, температура, относителна влажност на въздуха, осветеност, плътност на слънчева радиация.

Microprocessors microclimate parameters measurement system

Iliya I. Hadzhidimov, Irina P. Pavlova, Krystin K. Yordanov

Abstract: A microprocessor system for measurement and recording basic microclimate parameters like temperature, air relative humidity, solar irradiance and illumination is discussed. The functioning of the system can be independent due to opportunities for SD data recording without of a personal computer. By the data collected, the microclimate basic parameters can be analyzed and compared with the official requirements.
Keywords: microclimate parameters, microprocessors system, temperature, relative humidity, illuminance, solar irradiance.

1. Увод

Изискванията за комфорт и параметри на микроклимата в помещения с различно предназначение са обект на изследвания и анализ. Това е от съществено значение за реализация на мерки за енергийна ефективност и икономия на енергия, особено актуални в днешно време. Създадената микропроцесорна система е автономна, без необходимост от връзка с персонален компютър и има възможност за съхранение на измерените параметри на магнитен носител от рода на SD. Такава система може да се използва за събиране на данни, както и да се даде оценка на параметрите по отношение на нормите за микроклимат в обитаемите помещения за оптимизиране на енергийните нужди и реализиране на мерки за икономия на енергия.

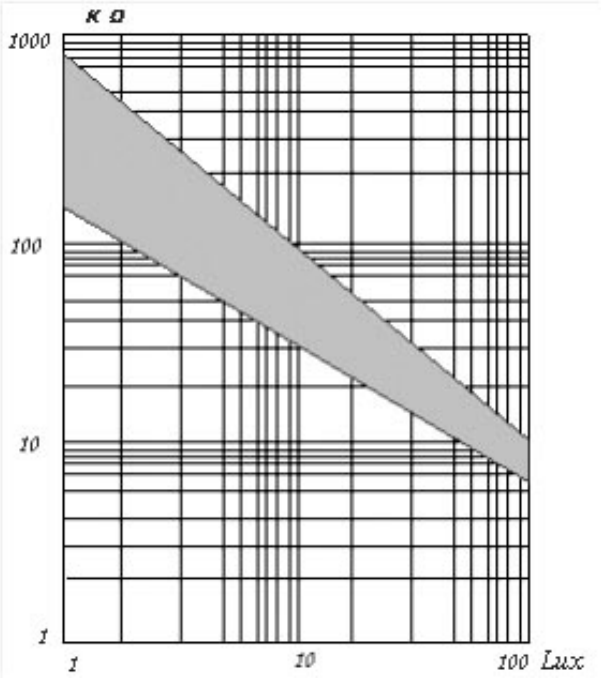
2. Микропроцесорна система на базата на ATMEGA328 и чувствителни елементи

За да се реализира измерване и съхранение на четирите параметъра, е използвана платформа ARDUINO [1] за микропроцесор ATMEGA328 с 10 битово аналогоцифрово преобразуване на сигналите. Платформата има достатъчен брой аналогови входове и цифрови портове, които могат да бъдат конфигурирани и като цифрови входове. За измерване на отделните параметри са избрани следните чувствителни елементи:

- Температура – цифров термометър DS18B20 [2]. Този термометър е един от най-използваните понастоящем сензори с точност, която задоволява напълно целта на измерването. В диапазона от -10^{o}C до +85^{o}C, точността му е +/- 0,5 ^{o}C. Има възможност за 9,

10 и 11 битово преобразуване на измерената температура. Максималната скорост на конвертиране на температурата в 12 битова дума е 750 ms. Друго съществено преимущество е възможността за използване на т.нар. „1-Wire" интерфейс, който използва само един цифров вход на микропроцесора. На една трикабелна връзка могат паралелно със захранване от ARDUINO или посредством т.нар. „паразитно захранване" да се включат до 1023 цифрови термометъра. Те са фабрично настроени и не е необходима допълнителна обработка на получената температура. Сигналът е цифров, което е предимство пред аналоговите сензори, които са чувствителни към външни електромагнитни полета.

- Относително влагосъдържание на въздуха – DHT21 [3]. Този цифров сензор е комбиниран за съвместно измерване на относително влагосъдържание и температура, при обхват от -40^{o}C до +80^{o}C точността на измерената температура е 0,1^{o}C, а за относителното влагосъдържание точността е +/- 3% (max 5%) при обхват от 0% до 100% относителна влажност (RH). Периодът на регистриране на показанията е средно около 2 секунди.
- Осветеност. За реализиране на измерването на осветеността е използван фоторезистор с аналогов сигнал от серията Ф5/GL55 [4] със съпротивление при липса на осветеност 3 MΩ. Чувствителността е от 30kΩ до 50kΩ за 10 lux, а времето за реакция е 20 ms при увеличаване и 30 ms при намаляване на осветеността. Работният температурен диапазон е от -30^{o}C до +70^{o}C. На фиг. 1 е представена зависимостта осветеност/електрическо съпротивление от производителя. За получаване на връзката между осветеност и електрическо съпротивление могат да се използват характеристиките, предоставени от производителя или да се извърши настройка на сензора с подходящ луксомер.



Фиг. 1. Зависимост между осветеност и електрическо съпротивление на Ф5/GL55/3 MΩ.

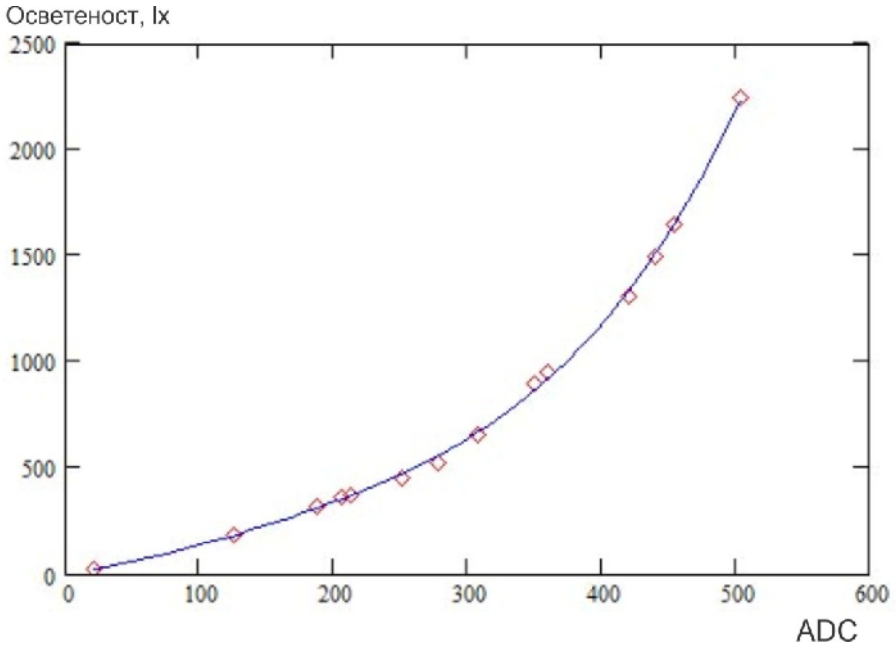
- Плътност на потока глобална слънчева радиация. Този параметър е важен за изучаване на т.нар. „пасивно слънчево отопление" през отоплителния сезон. Той дава възможност да се оцени каква част от слънчевата радиация, минаваща през прозорците и други прозрачни архитектурни елементи, попада в помещението и служи за пасивно отопление. На база на този параметър може да се направи анализ за възможни икономии на енергия за отопление. За измерване плътността на глобалната слънчева радиация е използван силициев фотоелемент с аналогов сигнал, който поради нелинейните си волт-амперни характеристики е необходимо да се настрои с еталонен пиранометър за получаване на точно

показание. Тъй като напрежението, получено вследствие на фотоволтаичното преобразуване е малко, за усилване на сигнала е използван операционен усилвател MCP602.

При дълги кабелни връзки, цифровият термометър DS1820 и сензорът за относително влагосъдържание се нуждаят от т.нар. “pull-up” съпротивление, което е необходимо за компенсация на увеличеното съпротивление от проводните линии.

2.1 Настройка на чувствителните елементи за преобразуване на аналоговите сигнали в изходна величина

Настройките са извършени за двата аналогови сигнала – осветеност и плътност на потока глобална слънчева радиация. Стойностите от аналогоцифрово преобразуване на фоторезистора Ф5/GL55/3MΩ са съпоставени при еднакви условия с показанията на цифров луксомер PeakTech 5025. На фиг. 2 е представена зависимостта на регистрираната с фоторезистора осветеност и показанията на луксомера.



Фиг. 2. Зависимост между показанието на аналогоцифровия преобразувател (ADC) и осветеността. Гладката линия е полином от 4-та степен с аргумент ADC.

За по-точно получаване на показанието за осветеност от фоторезистора, сравнението между аналогоцифровия сигнал и съответните стойности на PeakTech 5025 е ограничено до 2500 lux, които са достатъчни по изисквания за помещения с различно предназначение и са апроксимирани с полином от 4-та степен:

$$f(ADC) = 4{,}12 + 0{,}84.ADC + 6{,}495.10^{-3}.ADC^2 - 2{,}05.10^{-5}.ADC^3 - 4{,}35.10^{-8}.ADC^4. \quad (1)$$

Стандартното отклонение между измерените величини и полиномиалната функция е 20,6.

По същия начин се настройва и чувствителният елемент за измерване плътността на глобалния поток слънчева радиация.

2.2 Регистриране и съхранение на данните от измерване

С цел да се реализира микропроцесорна система, която може независимо от персонален компютър да дава възможност за измерване на данните, първична обработка и съхранение на показанията на магнитен носител е възприет следният подход:

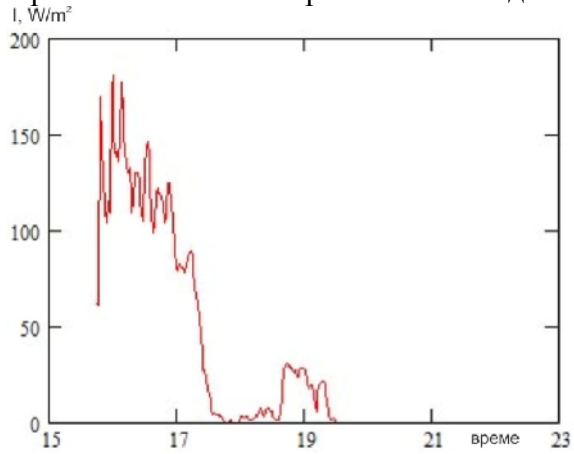
• ARDUINO платформата е снабдена с два т.нар. „Shield”, които по същество представляват надстройка на системата. Първият е „Sensor shield”, който позволява лесно и удобно да се включат четирите сензора. На тази надстройка всеки един от аналоговите и цифрови канали е с отделен пин за захранване, заземяване и отвеждане на сигнала. Вторият е т.нар. „Data logging shield” и е предназначен за самостоятелно записване на данните на Security Digital (SD) карта и има RTC (Real Time Clock) модул, с помощта на който, дори платформата да остане известно време без захранване, реалното време се съхранява и при рестарт на системата, данните се записват в реално време.

• За запис на данните е избран формат, който позволява следваща обработка на резултатите в различни програмни приложения за преглед и анализ. Данните се регистрират в стандартен “.txt” формат, като показанията от измервателните сензори се записват всяка минута. За да не се губи точност при резки промени в поведението на измерваните параметри, микропроцесорът е програмиран така, че в рамките на всяка секунда се запаметяват показания. След изтичане на една минута се пресмята средноаритметичната стойност за всеки параметър, който се записва на SD картата. Форматът на записване на данните е:

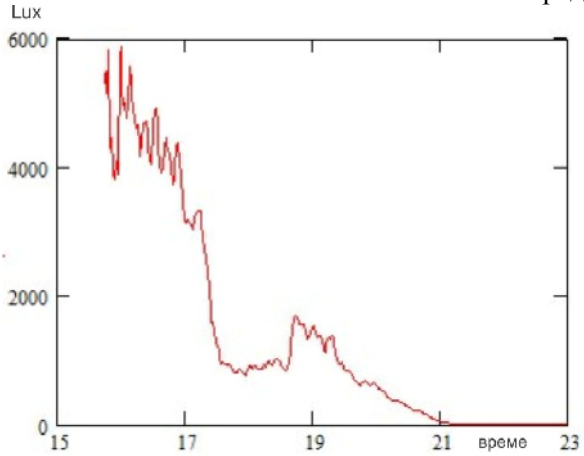
YYYY/MM/DD HH:MM:SS XXXXXXXXXX XXX.XX

Първият параметър (XXXX, целочислен) е плътността на потока слънчева радиация, вторият (XXXX, целочислен) е осветеността, третият (XX, целочислен) относителната влажност и четвъртият (XXX.XX, с плаваща запетая) е температурата.

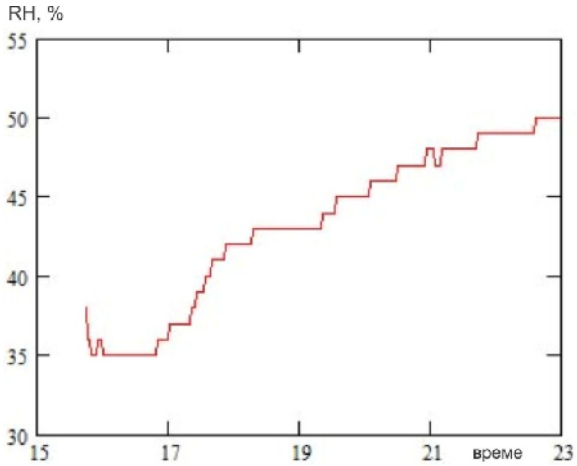
На фиг.3 а, б, в, г са представени графични резултати от направените тестове със системата. Тестовете са направени в стая в интервала от 15:45 до 23:30 на 1-ви юли 2013 г.



Фиг. 3а. Зависимост плътност на потока глобална слънчева радиация от времето



Фиг. 3б. Зависимост на осветеността от времето



Фиг. 3в. Зависимост на относителното влагосъдържание във въздуха от времето

Фиг. 3г. Зависимост на температурата на въздуха в помещението от времето

3. Изводи

Реализираната микропроцесорна система е компактна, функционална и успешно може да се използва за регистриране и записване на различни величини. Към системата могат да се включат и други чувствителни елементи, например за измерване на атмосферно налягане, още цифрови термометри и др., с което да се повиши нейната функционалност. Данните, получени от измерване, направено за голям период от време, могат да дадат представа за условията за работа или обитаемост в помещения с различно предназначение, а също така да се направи анализ за възможностите за пасивно отопление през зимните месеци, съобразно архитектурните и строителни особености на сградата. Недостатък представлява необходимостта от постоянно външно захранване на платформата. Стъпка напред би представлявало включване на UPS, с помощта на което да се подпомогне регистрация на данни през цялата година в случаи на прекъсване на електрозахранването. Ако е необходимо да се наблюдава и контролира системата, тя може да бъде допълнена с RF трансивери, един на ARDUINO модула и един на персонален компютър. По този начин данните могат както да се записват на SD, така също да се предават през виртуален COM по двойката трансивери по избрана честота, като такава двойка APC200 е тествана успешно на разстояние 1000 m свободно пространство.

Настоящата публикация е реализирана благодарение на научноизследователски проект 494-ПФ, финансиран целево от държавния бюджет за 2013 г.

Литература

[1]. ARDUINO, http://www.arduino.cc/, 2013.
[2]. DS1820 Programmable Resolution 1-Wire Digital Thermometer, Maxim Integrated, San Jose, CA, USA, 2008.
[3]. HM2301 Digital-output humidity and temperature sensor, Hanwei, http://www.hwsensor.com.
[4]. CdS Photoresistor Manual, GL55 Series, SHENZHEN SENBA OPTICAL & ELECTRONIC CO., LTD, www.sbcds.com.cn, 2013.

За контакти:
доц. д-р Илия Иванов Хаджидимов,
катедра „Топлотехника”,
Технически университет – Варна,
e-mail: i_hadzhidimov@tu-varna.bg.

АСИНХРОННО МИКРОКОНВЕЙЕРНО ЗВЕНО С ЦИФРОВ КОМПАРАТОР

Димитър С. Тянев

Резюме: Анализиран е процесът на превключване на комбинационната логическа схема на цифров компаратор. Въз основа на направените изводи е предложена нова и оригинална логическа схема за реализация на практически модел на фактическото закъснение при изпълнение на операция сравнение. Сигналът, който генерира синтезираната схема, позволява микроконвейерното звено да функционира в условията на асинхронен метод на управление.

Ключови думи: микроконвейерно звено, цифров компаратор, признаци *LT, EQ, GT*, закъснение, фактическо закъснение, асинхронно управление.

Completion detection model for micro-pipeline stage with digital comparator

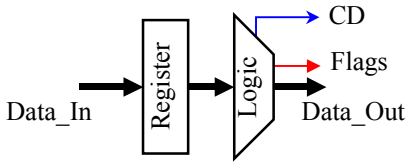
Dimitar S. Tyanev

Abstract: The process of switching a circuit of a digital comparator is analyzed. Based on the derived conclusions logic circuit is proposed for the realization of a new and original practical model of the completion detection in the execution of operation comparison. The signal generated by the synthesized circuit allows the asynchronous control method of the micro-pipeline stage.

Keywords: micro-pipeline stage, digital comparator, flags *LT, EQ, GT*, delay, completion detection model, asynchronous control.

1. Въведение

Настоящото изследване има за цел създаване на модел на фактическото закъснение (*Completion detection*) на микроконвейерно звено с цифров компаратор, показано на фиг. 1.



Фиг. 1. Логическа структура на микроконвейерно звено с цифров компаратор

Както е известно, операция сравнение генерира признаците (*Flags*) на отношението

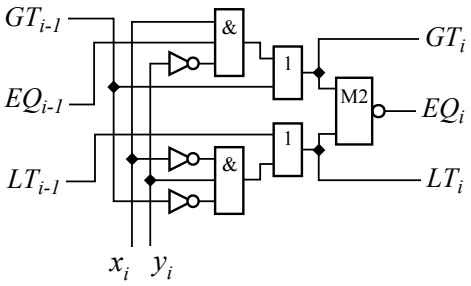
$$x * y, \qquad * \in \{<, =, >\}, \tag{1}$$

необходими при реализация на условни алгоритмични преходи.

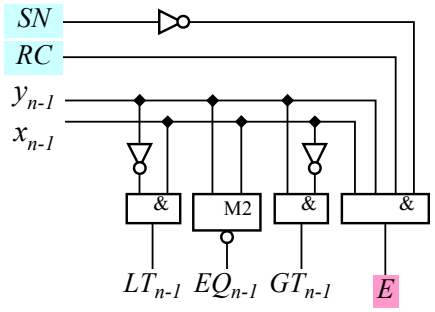
Признаците, които компараторът формира *LT (Less Than), EQ (equal)* и *GT (Greater Than)*, се дефинират както следва

$$\begin{aligned} &if \ \ (x < y) \ \ then \ \ LT = 1, \ EQ = 0, \ GT = 0; \\ &if \ \ (x = y) \ \ then \ \ LT = 0, \ EQ = 1, \ GT = 0; \\ &if \ \ (x > y) \ \ then \ \ LT = 0, \ EQ = 0, \ GT = 1. \end{aligned} \tag{2}$$

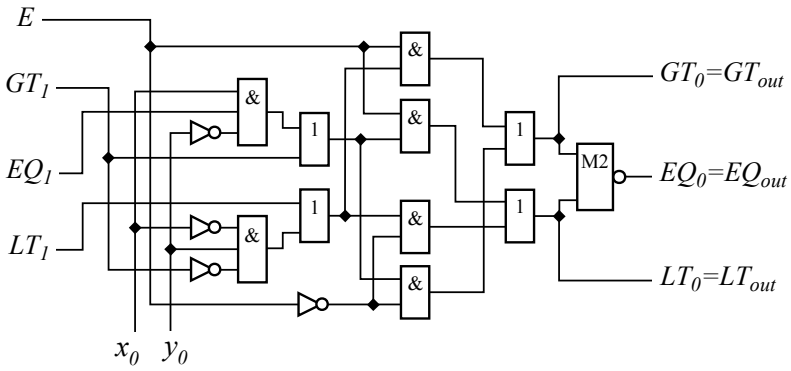
Тук се има предвид мултиформен и мултиформатен цифров компаратор, чийто синтез е изложен подробно в [1], а получените логически схеми на отделни разряди, са представени по-долу на фигури 2, 3 и 4.



Фиг. 2. Логическа схема на *i*-тия бит



Фиг. 3. Логическа схема на старшия бит



Фиг. 4. Логическа схема на младшия бит

За да е възможно микроконвейерното звено да функционира в условията на асинхронно управление, компараторът трябва да реализира модел на фактическото закъснение чрез сигнала *CD* (фиг. 1) Този сигнал следва да се използва в качеството на сигнал *Request*, предназначен за конвейерния автомат, който синхронизира изчислителния процес.

Модел на фактическото закъснение

Моделът на фактическото закъснение, предложен в [2], [3], се основава на логическа функция на паралелната парафазна дизюнкция (*dual-rail*) от логически сигнал *s*.

$$cd = s_{true} \cup s_{false} \ . \tag{3}$$

Този модел за реализация на фактическото закъснение обаче е само теоретичен и е не особено надежден на практика. Неговият недостатък се дължи преди всичко на факта, че на практика е невъзможно да се осигури абсолютната паралелност във времето на фронтовете в двете фази s_{true} и s_{false} при превключване на сигнала *s*. Отговорността за това е на технологиите, които не са в състояние да гарантират еднакво закъснение в различните вериги от последователно превключващи се логически елементи. Нещо повече, реализацията в чист вид на функция (3) се постига чрез двойно увеличаване на апаратните разходи, както може да се види в [4] и в [5].

Парафазната дизюнкция (3) теоретично е константа, но нейната практическа реализация генерира смущение, което се използва, за да се формира сигнал, идентифициращ началото и края на едно превключване. Същността и етиологията на този процес са изяснени в [4]. В по-общата интерпретация практическият ефект от тази функция може да се използва в качеството на сигнал за край на изчислението, провеждано от даден логически възел.

Логиката на тук разглежданата схема е такава, че тя като многоразрядна, формира закъснението на своето превключване последователно. С други думи, закъснението, с което цифровият компаратор формира изходните признаци, изразено чрез сигнал *CD* (фиг. 1), се формира като сума от закъсненията на последователните поразрядни латентности CD_i. Това разбиране се изразява чрез следната логическа функция:

$$CD = \bigcup_{i=n-1}^{i=0} CD_i \ , \tag{4}$$

където за сигналите CD_i тук се търси практическа реализация, основаваща се на (3).

За да се приложи в условията на многоразряден компаратор, предвид на изказаните по-горе съображения, моделът на паралелната парафазна дизюнкция следва да се реконструира.

Като се има предвид, че поразрядните сравнения продължават от разряд в разряд само след равенство в текущия разряд и се прекратяват след текущо неравенство, то според (2) следва, че сигналът EQ_i и дизюнкцията ($LT_i \cup GT_i$) имат винаги противоположни логически стойности. Съществено е да се отбележи още, че теоретично тези стойности възникват паралелно във времето. Основание за паралелността, изявена тук, са още и логическите схеми на компаратора, представени на фиг. 2, 3 и 4.

Въз основа на горната констатация се разкрива една нова възможност за реализация на фактическото закъснение. Избягвайки определението и основавайки се на алтернативността между двойката сигнали EQ_i и дизюнкцията ($LT_i \cup GT_i$), тук се предлага следната логическа функция

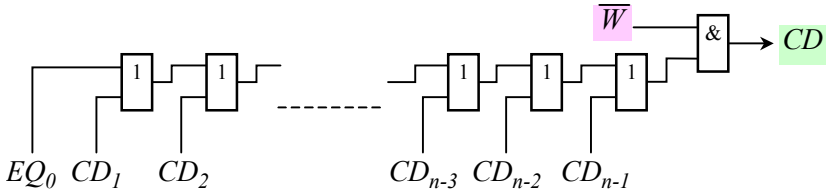
$$CD_i = \overline{EQ_i} \cup (LT_i \cup GT_i) \ . \tag{5}$$

Същността на функция (5) е такава, че тя гарантира логическа стойност нула при равенство в текущия разряд. Тази нула се конкатенира с всички следващи нули, до момента, когато редицата от последователни равенства се прекрати, което е в съответствие с (4). С прекратяване на тези последователни равенства, в поредния разряд възниква логическа единица като стойност на дизюнкцията ($LT_i \cup GT_i$), с което сравненията във всички следващи разряди стават безсмислени. Практически възникването на единица бележи във времето момента на получаване на крайния резултат.

В частния случай, когато сравняваните комбинации са еднакви, за маркиране на края на превключването на схемата следва да се използва стойността $EQ_0 = 1$, възникваща последна във времето. Така окончателно, за изходния сигнал *CD* тук се предлага следната логическа функция

$$CD = EQ_0 \cup \bigcup_{i=n-1}^{i=1} CD_i \tag{6}$$

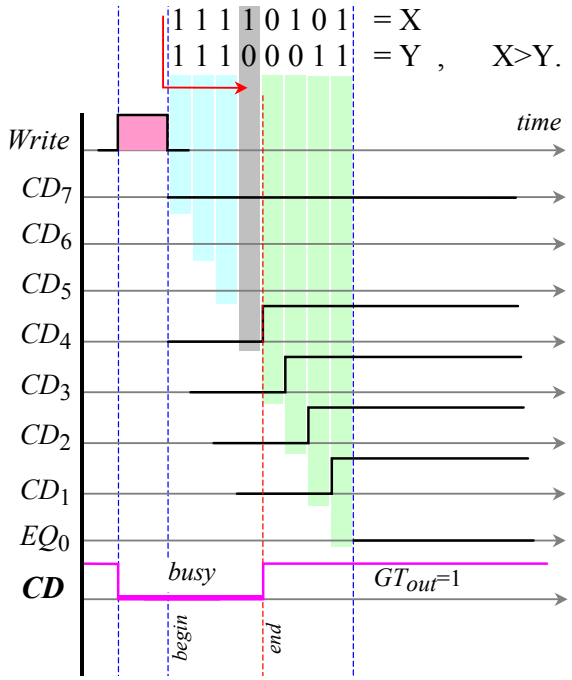
Хардуерната реализация на функция (6), представена на фиг. 5, се характеризира с изпреварващо във времето извеждане на резултата на изход, така щото ненужните превключвания могат само да потвърдят (при това със закъснение) вече получения резултат.



Фиг. 5. Логическа схема на сигнала *CD*

Включването на изходния логически елемент И се дължи на едно изключение. Последното се състои в това, че след текущо сравнение не е изключено появата като два нови операнда на същите комбинации. При това логическите стойности в тригерите на регистъра (фиг. 1) няма да се променят, а само ще се потвърдят. Това означава, че в компаратора няма да възникнат никакви превключвания, което ще осуети генерирането на сигнала *CD*.

Процесът на превключване на логическата схема от фиг. 5 и формирането на изходния сигнал, описан по-горе, е илюстриран с конкретен пример, представен на фиг. 6.



Фиг. 6. Времедиаграма на сигнала *CD* при сравнение на две 8-битови комбинации

3. Заключение

Поставената в началото цел е постигната. Синтезирана е принципно нова логическа функция (6), оценяваща продължителността на логическото изчисление на признаците на отношение (1). Хардуерната реализация на тази функция се основава на принципа за изпреварващо във времето извеждане на резултата, чието възникване се характеризира с променящо се местоположение по дължината на разрядната мрежа. При всяко сравнение функцията *CD* генерира преден фронт, което е необходимото условие за управление на конвейерния автомат. Нулевото ниво на тази функция съответства на продължителността на логическото изчисление. Може да се отбележи случайния характер на стойността на фактическото закъснение, чието изследване е представено в [1].

Литература

[1]. Тянев Д. С., Д. Г. Генов, Мултиформен, мултиформатен цифров компаратор, ТУ-Варна, “Компютърни науки и технологии”, ISSN 1312-3335, /под печат/.

[2]. Daves, Al, S.M. Nowick, An Introduction to Asynchronous Circuit Design, Technical Report UUCS-97-013, Computer Science Department, University of Utah, 1997.

[3]. Sparsø, Jens. Asynchronous circuit design - a tutorial, Technical University of Denmark, 2006.

[4]. Тянев, Д.С., Организация на компютъра. Цифрова аритметика, ISBN: 978-954-20-0258-0. Технически Университет - Варна, 2007.

[5]. Mocho, R.U.R., G.H. Sartori, R.P. Ribas, A.I. Reis. Asynchronous Circuit Design on Reconfigurable Devices, Proceedings of the 19-th annual symposium on Integrated circuits and systems design (SBCCI'06), August 28-September 1, 2006, Minas Gerais, Brazil, ACM: 1-59593-479-0, pp.20-25.

За контакти:
доц. д.н. инж. Димитър С. Тянев,
ТУ – Варна, катедра КНТ
www.tyanev.com .

РЕАЛИЗАЦИЯ НА TCP/SCTP WEB СЪРВЪР

Христо Г. Вълчанов, Димитър Н. Тодоров

Резюме: Голяма част от съвременните Интернет приложения се базират на надеждна доставка на съобщенията. Протоколът HTTP е пример за подобно приложение. Основните изисквания, които той налага към транспортния протокол, са: надеждна доставка на всеки файл и частична подредба на пакетите на всеки файл, но не и тотална подредба между всички пакети. Използваният транспортен протокол TCP предоставя много повече възможности, но същевременно внася значителни разходи при комуникацията. В доклада се представя реализация на web сървър, който използва транспортен протокол SCTP. Показана е архитектурата на сървъра и резултатите от проведените експериментални изследвания за оценка на неговата функционалност.
Ключови думи: Web server, TCP, SCTP.

Implementation of a TCP/SCTP Web Server

Hristo G. Valchanov, Dimitar N. Todorov

Abstract: Many of today's Internet applications are based on the reliable delivery of messages. The HTTP protocol is an example of such application. The main requirements that it imposes to the transport protocol are: reliable delivery of each file and partial ordering of packets of each file, but not a total ordering of all packets. The widely used transport protocol TCP provides many additional features, but also brings significant cost in communication. The article presents an implementation of a web server that uses the transport protocol SCTP. The architecture of the server and the results of experimental tests are presented.
Keywords: Web server, TCP, SCTP.

1. Въведение

Голяма част от съвременните Интернет приложения, както и разпространяващите се облакови услуги изискват надеждна доставка на съобщенията. Облаковите системи са изградени на базата на множество web сървъри, които са обединени в една обща мрежа и споделят ресурсите си. Протоколът HTTP е пример за подобно приложение [1]. Основните изисквания, които той налага към транспортния протокол, са: надеждна доставка на всеки файл и частична подредба на пакетите на всеки файл, но не и задължителна тотална подредба между всички пакети.

Използването на множество потоци (*multistreaming*) на транспортно ниво представлява възможността транспортните протоколи да поддържат множество потоци, като всеки от тях съдържа логическа последователност от данни със собствена подредба на пакетите. Във всеки поток данните се доставят до приложението независимо от реда на пристигане спрямо другите потоци. Тази характерна особеност прави потоците изключително подходящи за прехвърляне на независими web обекти. Когато всеки обект се предава в отделен поток, неговата обработка и визуализация не зависи от успешното предаване и доставка на други обекти. Текущо използваният протокол TCP не поддържа потоково предаване на транспортно ниво. По времето, когато е бил разработен, мрежовите приложения са имали критични изисквания към транспортните протоколи основно по отношение на задръстванията в мрежите и управлението на потока данни. В последствие, при появата на протокола HTTP, протоколът TCP се явява единствено решение сред съществуващите към момента транспортни протоколи. Същевременно обаче, предаването на независими web обекти върху TCP се отразява негативно на времето за отговор. Причините са две: първо,

TCP връзката предоставя на приложението единичен последователен поток от байтове и второ, осигурява поредна доставка в рамките на този поток от байтове. Ако една част от един обект не се получи поради загуба в мрежите, последователно предаваният обект няма да бъде доставен на клиента, докато тази част не бъде предадена повторно и доставена успешно. Въпреки предположенията за подобряване на времето за отговор при HTTP [2], не са известни достатъчно изследвания за доказателство на тази хипотеза.

В настоящия доклад се разглеждат някои аспекти на архитектурата и реализацията на HTTP сървър, използващ транспортните протоколи TCP и SCTP. Основната идея е разработване на функционален web сървър, който да се използва за експериментални изследвания с цел подобряване на трафика на мултимедийни страници през компютърни мрежи, изградени на базата на WAN технологии с ниска пропускателна способност.

2. Характерни черти на протокола SCTP

Протоколът SCTP е стандартизиран от IETF [3]. Той предлага възможности, аналогични на TCP- надежден сесийно ориентиран транспорт, гарантиращ реда на доставяне на данните. В допълнение, SCTP предоставя нови функционалности. За разлика от TCP, който е ориентиран към предаване на байтове, SCTP е базиран на съобщения. Сесията при него се нарича „асоциация". Създаването на асоциация изисква договаряне на 4 етапа (4-way handshaking), където данните могат да бъдат включени в третото и четвъртото съобщение на договарянето и да се изпратят, когато асоциацията е вече била валидирана. За защита от някои типове DoS атаки в процеса на договарянето е вграден и механизмът на cookie.

Едни от най-характерните черти на SCTP са многопотоковото предаване (multistreaming) и поддържането на множество адреси на хост (multihoming). Чрез многопотоковото предаване предаваните в рамките на една асоциация данни могат да се разделят в няколко отделни потока, всеки поддържащ собствена последователност на доставяне. За всеки поток може да се конфигурира подредена или неподредена доставка. Загубените пакети от един поток не влияят върху останалите потоци. Това позволява елиминиране на т.н. „TCP Head of Line Blocking" проблем. Същевременно, предаването се организира в рамките на една асоциация, така че всичките потоци са обект на общ механизъм за управление на обмена на данни и задръстванията в мрежата.

Поддържането на множество адреси дава възможност всяка крайна точка на SCTP асоциация да има множество IP адреси, като за целта се обменя списък с тях по време на процеса на договаряне. За всяка SCTP страна се използва единичен номер на порт по отношение на тези адреси. Само един от тях е активен и се използва за обмен, докато останалите осигуряват резервираност в случай на отпадане на активния IP адрес. Тази особеност на протокола осигурява висока надеждност на обмена чрез поддържане на множество маршрути от тип край-до-край.

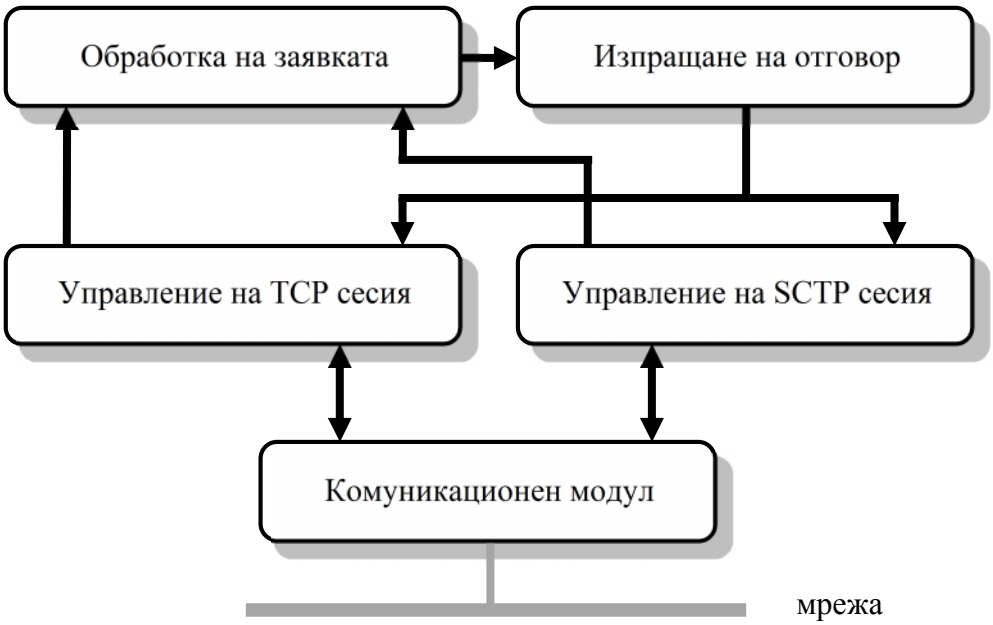
3. Архитектура на TCP/SCTP web сървър

Мотивацията за разработване на web сървър изцяло от началото е продиктувана от факта, че е известна само една разработка с отворен код на подобен сървър, която е базирана на Apache [4,5]. Този проект включва адаптиране на ядрото на сървъра – APR (Apache Portable Runtime) за поддръжка на SCTP, както и модификация на браузъра Firefox. В настоящия момент обаче този проект е в застой поради финансови причини.

При проектирането на архитектурата на сървъра са взети предвид следните съображения:

- да не се внасят промени в съществуващата спецификация на протокола HTTP;
- сървърът да има модулна архитектура, позволяваща бъдещо развитие.

Текущата реализация позволява сървърът да функционира в режим на обмен само с TCP или само с SCTP клиенти. Модулната архитектура обаче дава възможност бъдещите реализации да подържат едновременно функциониране и в двата режима. На фиг.1 е показана архитектурата на сървъра. Той е реализиран на езика C под Linux 2.6.



Фиг. 1. Архитектура на сървъра

Модулът за обработка на заявката анализира получената заявка. Определя се изисквания обект, местоположение му във файловата система и неговата дължина. Обектът се извлича и се формира отговор.

Модулът за изпращане на отговор подготвя връщаните данни в подходящ формат. Ако е настъпила грешка по време на обработката на заявката, този модул формира отговор, съдържащ информация за грешката.

Функционалността на модула за управление на TCP сесия се базира на системната библиотека на Linux и се заключава в:

Установяване и затваряне на сесията.

- Потвърждаване и избягване на задръствания – механизми, чрез които се откриват пропуски в предаваните съобщения и се регулира потока от данни.
- Валидиране на данни – използва се полето контролна сума за проверка дали не са били повредени данните по време на прехвърлянето.
- Подредено доставяне – при прехвърлянето на данни е възможно те да пристигнат в разбъркана последователност. Този механизъм се грижи при доставяне данните да бъдат правилно подредени.
- Фрагментиране – когато прехвърляните данни имат твърде голям обем, те се разбиват на отделни фрагменти при изпращане и след това се сглобяват при получаване.

Компонентът за управление на SCTP сесия е разработен на основата на прототипната библиотека sctplib-1.0.11 [6]. Неговото предназначение включва:

- Подредено доставяне в рамките на потока – за всеки поток индивидуално се следи дали данните пристигат подредени. Загубата на данни в даден поток не оказва влияние върху другите потоци.
- Установяване и затваряне на асоциацията – включва механизми за създаване, нормално затваряне и прекъсване на асоциация.

- Пакетиране на данни – в рамките на едно SCTP съобщение могат да са пакетирани няколко фрагмента данни.
- Фрагментиране на потребителските данни – когато прехвърляните данни имат твърде голям обем, те се разбиват на отделни фрагменти при изпращане и след това се сглобяват при получаване.
- Потвърждаване и избягване на задръствания – механизми, чрез които се откриват загуби в предаваните съобщения и се регулира потока от данни.
- Валидиране на данни – използва се полето контролна сума за проверка дали не са били повредени данните по време на прехвърлянето.

Предназначението на комуникационния модул е да реализира достъпа до комуникационните примитиви на операционната система.

В режим на функциониране като TCP сървър, при постъпване на нова заявка от страна на клиента за създаване на конекция, сървърът стартира отделна нишка, която обслужва тази заявка. Това дава възможност за обработване на едновременни заявки от множество клиенти. След формирането на отговор, работната нишка завършва.

В режим на SCTP сървър има определени разлики. SCTP сървърът е еднонишков, поради ограниченията на прототипната версия на библиотеката, което се дължи на проблем при синхронизацията. Този проблем се предвижда да се реши в бъдещи реализации. Всяка една създадена асоциация има множество входни и изходни канали, които се създават при нейното иницииране. Каналът, по който сървърът предава данните, трябва да е същия по който ги е получил. При SCTP реализацията не се извършва проверка дали получената заявка е минала през прокси сървър или не, поради отсъствието на средства за определяне на директна връзка от браузъра към сървъра в библиотеката. Това също може да бъде една насока за бъдещо развитие.

4. Експериментални изследвания и резултати

Целта на експерименталните изследвания е да се установи функционалността на сървъра и да се изследва поведението на двата транспортни протокола в мрежова инфраструктура, изградена на базата на WAN технологии. Този тип технологии осигуряват пропускателна способност в порядък по-малка от технологиите за локални мрежи. Това е съществен фактор, който влияе върху времето за отговор на web сървъра в глобален мащаб.

Експерименталната постановка е показана на фиг.2. Тя включва маршрутизатори Cisco 2901, VLAN комутатори Cisco Catalyst 2960, компютри HP Desktop 500B CPU Intel Core Duo E5800 3,2GHz, 2G DDR3 RAM. Платформата за web сървъра е базирана на Slackware Linux 2.6 и gcc 4.4.3. За клиента е използвана ОС Windows 7 с браузър Google Chrome 28.0.1500.

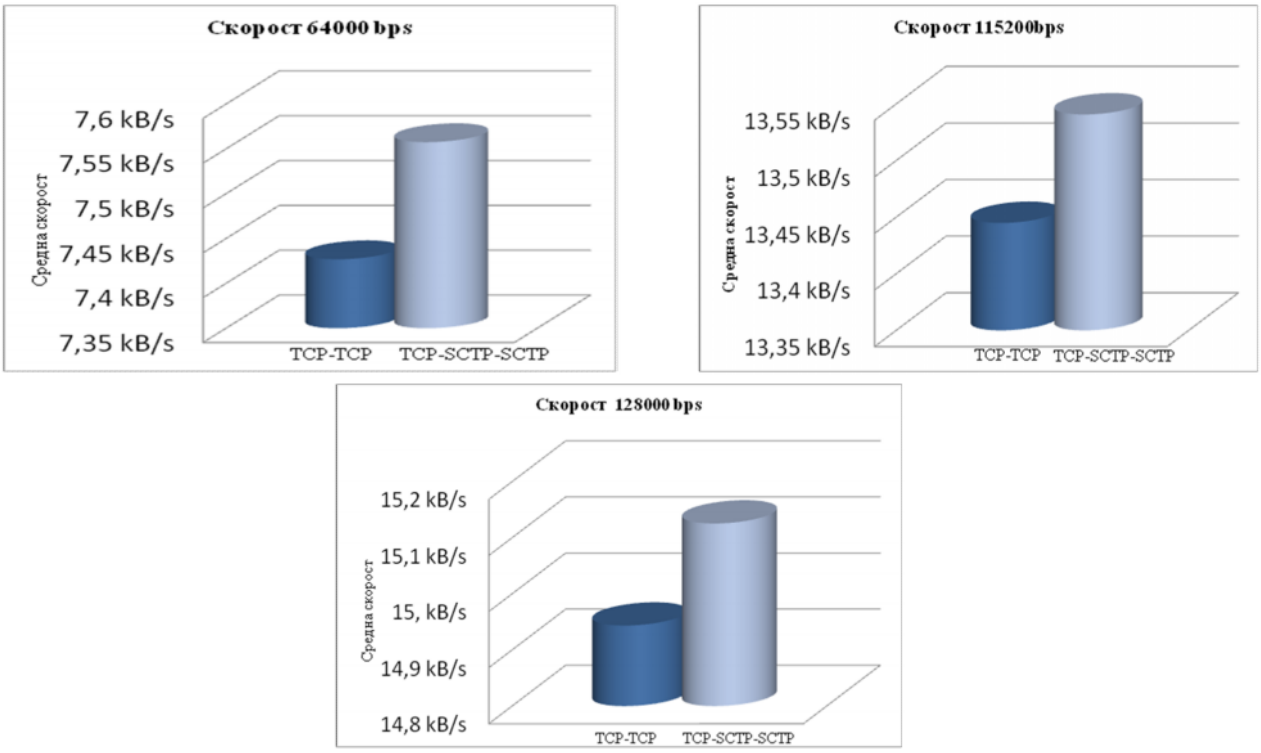
Между маршрутизаторите са изградени серийни връзки на базата на протокола HDLC. Чрез тяхното конфигуриране с различни скорости се симулира WAN преносна среда с различна пропускателна способност. Експериментите са извършени в две насоки:

- комуникация между TCP клиент и сървъра, работещ в TCP режим (фиг. 2а).
- комуникация между TCP клиент и сървъра, работещ в SCTP режим (фиг. 2б). В този случай се използва междинен TCP/SCTP прокси сървър, който не се разглежда в настоящата публикация.

Фиг. 2. Експериментална постановка

Експериментите са направени при различни скорости на серийната връзка. Клиентът изтегля мултимедийна страница с размер 300Kb, съдържаща различни обекти, намиращи се на същия сървър. Извършени са определен измервания и са отчетени средните скорости на зареждане на страница.

На фиг. 3 са показани резултатите от проведените изследвания.



Фиг. 3. Резултати от изследванията

Получените резултати дават основание да се заключи, че разработеният сървър е функционален и може да се използва като експериментално средство за оценка на предаването на мултимедийни данни. На второ място, експериментално се доказа предимството на протокола SCTP пред TCP по отношение на ефективността му при мрежови инфраструктури, изградени на базата на WAN технологии. Независимо от скоростта на връзката, протоколът SCTP позволява постигането на по-добри времена на зареждане. Следва да се отбележи, че за прилагането на SCTP се използва допълнителен TCP/SCTP прокси сървър, който неминуемо внася допълнително времезакъснение при обмена на данните.

По-добрите времена, които се получават при зареждане на страница с използване на SCTP, се дължат на общия му механизъм за контролиране на потока, който намалява количеството използвана служебна информация, в сравнение с TCP, когато се предават множество ресурси паралелно. Също така, зареждането на страницата се извършва в рамките

на една SCTP сесия, за разлика от TCP, където за всеки ресурс се създава нова сесия. Като допълнение може да посочи, че и двата протокола осигуряват надеждна доставка на данните, без загуба на информация.

5. Заключение

В настоящия доклад е представена реализация на експериментален TCP/SCTP HTTP сървър. Сървърът притежава модулна архитектура, което позволява негово бъдещо развитие. Проведени са експериментални изследвания в реална мрежова инфраструктура. Получените резултати показват функционалността на разработения сървър, както и предимствата на протокола SCTP пред аналогичния TCP при обмен на мултимедийни данни през мрежи, изградени на базата на WAN технологии.

Като насоки за бъдеща работа могат да се посочат въвеждане на нишки за обслужване на множество клиенти едновременно, както и разширяване на функционалността на сървъра с възможността да обслужва едновременно TCP и SCTP сесии.

Литература

[1]. Hypertext Transfer Protocol - HTTP/1.1, http://tools.ietf.org/html/rfc2616.
[2]. Gettys J., Email to End2end-interest Mailing List, October, 2002.
http://www.postel.org/pipermail/end2end-interest/2002-October/002436.html.
[3]. An Introduction to the Stream Control Transmission Protocol (SCTP),
http://tools.ietf.org/html/rfc3286.
[4]. Natarajan P., J.Iyengar, P.Amer. SCTP: An Innovative Transport Layer Protocol for The Web. Proc. of 15th Conf. of WWW, 2006, pp.23-26.
[5]. SCTP apache server: http://www.eecis.udel.edu/~leighton/firefox.html.
[6]. SCTP library: http://www.sctp.de/sctp.html.

За контакти:
гл. ас. д-р инж. Христо Г. Вълчанов
катедра „Компютърни науки и технологии”
Технически университет - Варна
hristo@tu-varna.bg

инж. Димитър Н. Тодоров
dntodorov@hotmail.com

МЕТОДИ ЗА ВЪЗСТАНОВЯВАНЕ НА ПЪТ В MPLS МРЕЖИ

Венета П. Алексиева

Резюме: Основната концепция на Multi-протокол Label Switching (MPLS) мрежа се базира на Label Switch Path (LSP) техника, която осигурява висока производителност при доставянето на пакетите, т.к. игнорира проверка в маршрутните таблици на устройствата. При отпадане на връзка, MPLS технологията я възстановява, намирайки нов път, но при този процес понякога част от трафика с нисък приоритет може да се загуби. В този доклад се предлагат два метода за възстановяване на път, които установяват втори LSP за връзки, при които натоварването надхвърля 90% и се появява възможност за загуба на пакети. Резултатите показват, че предложените методи подобряват QoS, т.к. намаляват загубата на пакети с по-нисък приоритет по претоварените връзки по LSP.
Ключови думи: MPLS, възстановяване на LSP, протокол за възстановяване в MPLS мрежа.

Methods for path recovery in MPLS networks

Veneta P. Aleksieva

Abstract: The basic concept of Multi-Protocol Label Switching (MPLS) network uses Label Switch Path (LSP) technique that provides high performance in the delivery of packages, because ignores examination of devices routing tables. When link faults, MPLS technology recover it, finding a new path, but during this process sometimes part of low-priority traffic may be lost. This paper proposes two methods for LSP recovery that sets a second LSP for connections where the load exceeds 90% and appears the possibility of packet loss. Simulation results present that the proposed methods improve the QoS, due to minimization of the loss of packets with lower priority in congested links on LSP.
Keywords: MPLS, path recovery, recovery protocol in MPLS network

1. Увод

В наши дни Multi протокол Label Switching (MPLS) се утвърди като най-използваната технология за доставчиците на интернет услуги, защото позволява пренос на данни, глас и видео през една и съща мрежова инфраструктура и притежава редица предимства като управление на трафика чрез приоритети, високо качество на услугите (QoS), гъвкавост при избора на маршрути, възможност за реализиране на виртуални частни мрежи и мащабируемост на рутирането. MPLS технологията е подходяща за големи мрежи, със скъпо и специализирано оборудване, т.к. осигурява надежност, която се постига с два метода – защита и възстановяване. Тъй като възстановяването отнема по-малко мрежови ресурси и е по-евтино решение, отколкото защитата, се използва по-често от Интернет доставчиците на услуги. В настоящия доклад се изследват два предложени от автора метода за възстановяване в MPLS мрежи, при които се минимизира загубата на пакети от по-нископриоритетния трафик, като чрез симулации те са сравнени с класически методи за възстановяване.

2. Изследвания на отпадането на трафик при възникване на инцидент в MPLS мрежи

В реална ситуация, всеки един от предоставените ресурси в дадена мрежа може да предизвика грешка при предаването на пакети. Този инцидент поражда преизчисляване на маршрути, което отнема време, за да се превключи трафикът от пътя с повреда по алтернативен работещ път. През този период е възможно отпадане на голям брой пакети, като този брой зависи от редица фактори като: моментното натоварване по съответната

основна връзка; конфигурирания метод за възстановяване; избраната от администратора на мрежата грануларност на приоритетите, зададени за различните видове трафик по връзката.

Главният проблем се поражда от факта, че трафикът ще продължава да се изпраща по основния, но отпаднал път, докато информацията за инцидента достигне това устройство, което трябва да прехвърли трафика по алтернативния път. В този период от време пакетите ще бъдат загубени, докато устройството, което отговаря за възстановяването на път, не получи информация, че има повреда и трябва да спре изпращането им по повредения път. Ако повредата се намира далеч от точката за възстановяване и скоростта на предаване е висока, броят на отпадналите пакети може да бъде много голям.

През предходните години изследванията на доставчиците на Интернет услуги се насочват към маршрутизиращи алгоритми с няколко пътя за маршрутизиране [8,11]. Изследванията показват, че е възможно да се предостави набор от алтернативни пътища в мащабируема мрежа, както с протоколи от трети слой на OSI модела, така и при MPLS. Предварителното (статичното) им задаване намалява времето за прехвърляне на трафика по алтернативния път, т.к. не се заделя време за динамично изчисляване на резервен маршрут. Проблемът при възстановяването обаче е, че решението коя връзка ще бъде алтернативна при отпадане на основна връзка е субективно и не винаги оптимално. В някои случаи при предлаганите решения за динамично пренасочване на трафика при възстановяване на път в MPLS мрежа се оказва, че след прехвърляне на трафика по резервната връзка, тя няма достатъчен капацитет да поеме целия трафик и част от пакетите отпадат и не се доставят. Затова се търсят различни методи за възстановяване на път, които да оптимизират настоящите решения в различни аспекти.

3. Класически методи за възстановяване в MPLS мрежи

Защитата на данновите потоци в случай на пропадане на връзка или маршрутизатор е много важна, особено за услуги в реално време и мултимедийни приложения. От решаващо значение е времето за възстановяване (*TR*), което представлява времето от възникване на проблем до възстановяването на напълно работоспособно състояние на мрежата. То се изчислява като:

$$TR = T_{fd} + T_n + T_{bc} + T_{sw}, \tag{1}$$

където: T_{fd}- време за откриване на инцидента, T_n- време за съобщаване на LSR за инцидента, T_{bc}- време за избиране на алтернативен път, T_{sw}- време за прехвърляне на трафика по алтернативния път.

За да се приложи ефективно технологията MPLS, независимо от метода на възстановяване, трябва винаги да се съблюдава изискването:

$$TR <= 60 ms \tag{2}$$

Търсените решения са в посока намаляване на това време, като най-често срещано решение е намаляването на *TR* чрез задаване на статични LSP, за които

$$Tbc = 0 \tag{3}$$

В [7] се предлага глобално възстановяване със защитно превключване, което предлага предварителна настройка на възстановяващия път и динамично възстановяване. В [5] се предлага защита от типа 1:1, прилагайки я глобално или за сегмент и използвайки „дърво за известяване”. В [4] идеята на обратната връзка е, че в момента на инцидент по работния път, вместо да се изпраща сигнал за известяване за отпадане на пътя и трафикът да продължи да се праща по неработещ път и да се губят пакети, самият трафик се връща обратно към точката за възстановяване. В този метод могат да се приложат и двете защити 1:1 и 1:N. Недостатък е, че докато устройството, което ще възстановява получава върнат трафик,

пакетите ще продължават да се пращат до точката на повреда по повредения път. Чак когато то получи върнатия трафик, започва да препраща входящия трафик по глобалния резервен път. Така за кратък период входящият трафик се смесва с върнатия до превключването му по резервния път. Затова при [3] се доразвива идеята на [4], като се предлагат два метода - на глобално последователно преизчисляване на път и на избирателно последователно преизчисляване на път. В изследването се предлага с 28% по-ниска цена за заделени възстановителни пътища в сравнение с решението на [4]. Друго подобно решение се прилага при [6], където когато първият пакет сработи като сигнал за известяване и пристигне в точката на превключване на посоката към алтернативния път, маршрутизаторът, който е в роля на точка за възстановяване, буферира получените следващи пакети, принадлежащи на същия основен път, а последният пакет, който маршрутизаторът изпраща по повредения път е отбелязан с установяване на бита в EXP полето в MPLS заглавната част.

Всички тези методи реализират възстановяване на път в MPLS мрежата в изискването от стандарта време (<60 msec), но винаги в периода на възстановяване се получава известна загуба на пакети. Нито един от тях не анализира натоварването по алтернативния път, а просто прехвърля трафика по него, което означава, че при пикови натоварвания по алтернативния път пропускателната способност може да не бъде достатъчна за целия обем трафик и да отпаднат пакети от по-нископриоритетния трафик, въпреки че алтернативни на този път пътища съществуват и подобни загуби могат да бъдат преодолени.

4. Предлагани методи за възстановяване в MPLS мрежи

Предлаганите тук методи следят натоварването на връзките в мрежата. Идеята е, че ако при възникване на инцидент след прехвърляне на трафика по статично конфигурираната алтернативна връзка тя няма достатъчно капацитет за двата трафика, вместо да се изгубят пакетите, които са с по-нисък приоритет и са над капацитета на връзката, те ще се премаршрутизират по втора алтернативна връзка, която се изгражда временно и динамично в момента на претоварването след инцидента. В тези случаи е възможно да се приложи различен механизъм за възстановяване на път за трафика с различен приоритет. Например, да се изпраща само трафика с приоритет 0 и 1 по основния алтернативен път, който е конфигуриран предварително (статично), докато общия трафик, който не е с толкова висок приоритет (>2) и по време на претоварване на връзката би бил изгубен, да се изпрати по път, който се изчислява динамично в момента на откриване на инцидента. При наличие на такова решение се очаква намаляване на натоварването на входно-изходните процесори на маршрутизатора, както и обема на използваната памет.

Идеята на тези методи е да се ползва техника за възстановяване при няколко едновременно възникнали инцидента, подобна на предлаганата в [10] техника Modified Flexible MPLS Signaling (MFMS), която осъществява възстановяването в пет фази: Установяване на LSP; Прехващане на инцидента; Възстановяване; Координация между входния маршрутизатор и наново установения път; Прекратяване на възстановяването. За първата от фазите се използва представения сигнален протокол в [9], който се предлага за премаршрутизиране в динамична мрежова среда и възстановява проблем с устройство или връзка локално. Първият от предлаганите методи се състои от две фази:

- **Off-line фаза** – По време на тази фаза се оптимизират статичните пътища - едновременно основните и защитните маршрути, като се използва и оригиналния RSVP-TE протокол. Главната цел на този етап е да се минимизира претоварването, когато възникне проблемът с пропаднала връзка или устройство.
- **On-line фаза** – През време на тази фаза след възникване на проблема и премаршрутизирането на трафика по алтернативния (възстановяващ) път,

маршрутизаторът прилага защитен механизъм, който избира динамично нов път за пакетите с по-нисък приоритет.

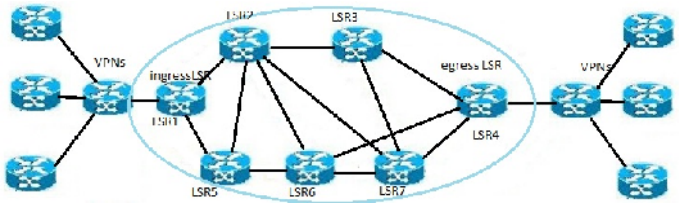
След като бъде открит инцидент, се преминава към прехвърляне на трафика по предварително установената резервна връзка. Съгласно този метод, описан по-подробно в [1,2], ако буферът на първия маршрутизатор по алтернативния път е запълнен над определен праг (който за целите на изследването се приема 90%), от него се стартира и алгоритъма за търсене на втори алтернативен път, който ако съществува, ще послужи за прехвърляне на по-нископриоритетния трафик. Така броят загубени пакети в момента на превключването по алтернативния път се получава същия като при защитно превключване. Преценката е субективна – на база на резултатите от редица симулации избраният критерий за задействане на динамичната част на алгоритъма по възстановяване е 90% запълване на буфера на първия маршрутизатор, открил инцидента. Ако тази стойност не е достигната в момента на прехвърлянето на трафика, алгоритъмът не се стартира.

Предприето е и изследване за олекотяване на този метод, като вместо за всеки пакет, е дадена възможността да се избере алгоритъмът по възстановяване да се активира един път за няколко пакета, т.к. подобно решение е интересно за мрежови приложения, които могат да покриват различни нива на качество на поддържаните услуги, когато допускат да понасят загуба на повече от един пакет за всяко приложение на алгоритъма. Така предлаганият тук втори метод усъвършенства предходния като следи динамично натоварването на връзките във всички маршрутизатори, независимо дали инцидент е възникнал и при преминаване на натовареността над избран праг (90% запълване на буфер на маршрутизатор) се стартира процедурата по търсене на алтернативен път за прехвърляне на по-нископриоритетния трафик по него, за да се предотврати евентуална загуба на пакети при недостатъчен капацитет на връзката. Методът осигурява и възстановяване на път при инцидент, като процесът по възстановяване след прехвърляне на трафика по защитната връзка не приключва с това, а се проверява натовареността на тази резервна връзка и се търси алтернативен път преди да е отпаднал трафик. Оптималността при търсенето на втори алтернативен път и тук касае определено ниво на качеството на услугите. Тъй като целта на намирането на втори резервен път е да не се губят пакетите с по-нисък приоритет, за които по първия резервен път е заделен достатъчен капацитет, се избира цикъл (последователност от връзки, формиращи нов път до същата дестинация), в който няма припокриване на връзки с връзките от първоначалния алтернативен път, по който вече се пренася и прехвърления трафик (защото именно те ще се явят като „тесни места” и ще предизвикат загуба на пакети от претоварване на връзката).

5. Експериментални изследвания и получени резултати

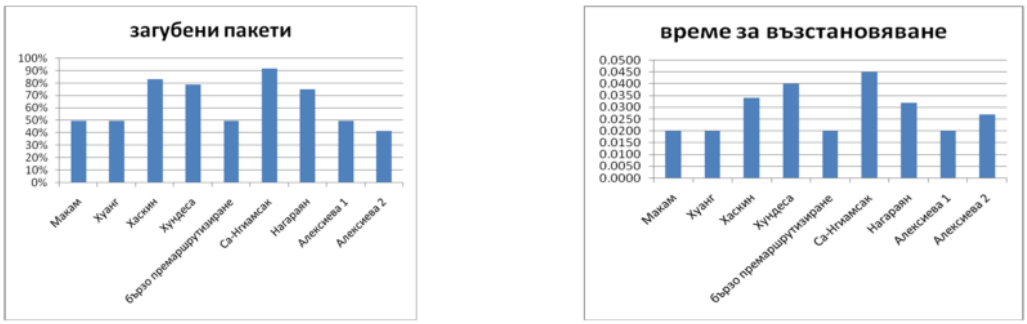
Според [12], моделът на симулация винаги се абстрахира от реалния свят, не е скъп, резултатите се получават бързо без да влияят върху реалната система, чието поведение се изследва, но също така резултатите не може да се прехвърлят директно върху нея, т.к. има характеристики на реалната система, които не могат да бъдат включени в модела, но също влияят върху резултатите. Обикновено се получават разлики в резултатите, получени от различните средства за реализация на симулации с еднакви параметри, което доказва, че за да бъдат резултатите между различни методи при едни и същи параметри сравними, трябва да се използва едно и също средство и да се работи върху един и същ модел. За симулациите в настоящото изследване е използван NS- 2.

За да се изследват предложените методи чрез симулация и резултатите да бъдат максимално близки до резултати, получени в реална система, е избран експериментален модел (фиг. 1), описващ връзките в реална MPLS мрежа на Vivacom в гр. София. От гледна точка на конфиденциалност е разгледано състоянието на тази мрежа към 2011г.



Фиг. 1. Експериментална MPLS мрежа

Тъй като при предходни изследвания на автора, представени в [2] резултатите показват, че не във всяка мрежа може да се намери втори алтернативен път, както и в различните мрежи броят на циклите, чрез които се намира алтернативен път, значително варира, тук са разгледани само варианти, при които съществува втори алтернативен път. Получените резултати са на база реално измерени изходни данни за трафика, пренасян с MPLS в тази мрежа. Направено е сравнение между различните методи за възстановяване за представената експериментална мрежа при едни и същи входни данни. На фиг. 2 е представено сравнение на получените резултати за брой загубени пакети при предложените и класическите методи и време за възстановяване в милисекунди.



Фиг. 2. Загубени пакети по време на възстановяване и време за възстановяване

Така представените показатели са включени при изследване на комплексните оценки на методите, като допълнителни критерии са: възможност за възстановяване при повече от един инцидент в мрежата; време за възстановяване при 2 инцидента; време за възстановяване при 3 инцидента. В таблица 1 са представени комплексните оценки на методите – аритметична (*Ra*) и геометрична (*Rг*).

Таблица 1. Комплексни оценки на методите

	Макам	Хуанг	Хаскин	Хундеса	Бързо премаршру тизиране	Са-Нгиамсак	Нагараян	Алексиева 1	Алексиева 2
Ra	0.477	0.352	0.693	0.301	0.685	0.311	0.266	0.690	0.700
Rг	0.323	0.327	0.451	0.297	0.527	0.436	0.397	0.464	0.554

Времето за възстановяване зависи от начина по който съответният метод изчислява съгласно формула (1) отделните времена. Именно по тази причина премаршрутизиращите механизми имат по-високи стойности спрямо защитно превключване, където резервният път е предварително известен, защото времето включва и времето за преизчисляване на пътя. Представените по-горе методи предлагат по-кратко време за възстановяване в сравнение с другите, тъй като първият резервен път е предварително известен и не се заделя време за преизчисляване на пътя. Пакетите започват да се доставят по новия път, докато тече времето на преизчисляване на друг резервен път за пакетите с по-нисък приоритет (ако капацитетът на резервния път е запълнен след прехвърляне на трафика над определен праг, който за целите на експеримента се приема 90%). Съгласно получените резултати може да се направи и изводът, че предложените методи са с много близки показатели спрямо най-добрите методи. Вторият предложен метод е с по-добри показатели по отношение на загубата на пакети спрямо първия предложен метод.

6. Заключение

В доклада са разгледани два метода за възстановяване на път в MPLS мрежа, които минимизират загубата на пакети от по-нископриоритетния трафик при възниване на инциденти в пикови моменти на натоварване на мрежата. Ако се приложи един от тук предложените методи за избор на алтернативен път в on-line фаза би довело до следните положителни резултати:

- Те осигуряват такива пътища, при които връзките могат да поемат трафика както при пикови натоварвания, така и допълнителния трафик, прехвърлен от пътища с отпаднали връзки.
- Те намаляват броя на отпаднали пакети от по-нископриоритетния трафик от пътя, използван като алтернативен на отпадналата връзка.
- Те ефективно премаршрутизират по-нископриоритетния трафик по пътища с несъвпадащи връзки с първия алтернативен път (ако такива бъдат открити).
- Те осигуряват пълнота и коректност на резултатите (получените пътища).
- Те водят до по-висока производителност на мрежата вследствие на по-малък брой пакети в буферите на маршрутизаторите и по-малък брой загубени пакети.

Литература

[1]. Алексиева В., Алгоритми за възстановяване в MPLS мрежи, моделирани с мрежи на Петри, UNITECH'12, 16–17 November 2012, Publ."V.Aprilov", Gabrovo– с.467-472.

[2]. Алексиева В., Метод за възстановяване на път в MPLS мрежи, John Atanasoff Society of Automatics and Informatics, Sofia, October 3-5, 2012, pp.337-340.

[3]. Cotter, R., D. Medhi. Survivable design of reconfigurable MPLS VPN networks. In: Proc. of Design of Reliable Communication Networks'2009, 2009, p.78 –85.

[4]. Haskin, D., R. Krishnan. A method for setting an Alternative Label Switched Path to Handle Fast Reroute. http://tools.ietf.org/html/draft-haskin-mpls-fast-reroute-01.

[5]. Haung, C., V. Sharma, K. Owens, V. Makam. Building Reliable MPLS Networks Using a Path Protection Mechanism. // IEEE Commun. Mag., 2002, vol. 40, issue 3, p. 156-162.

[6]. Hundessa, L., J.Pascual. Fast rerouting mechanism for a protected label switched path. In: Proc. of the IEEE International Conference on Computer Communication 01, 2001, p.527-530.

[7]. Makam, S. Protection/Restoration of MPLS Networks. http://zinfandel.levkowetz.com/ html/draft-makam-mpls-protection-00.

[8]. Marco, T., K Wu, A.Fumagalli, J.Vasseur. Local Detection and Recovery from Multi-Failure Patterns in MPLS-TE Networks. In: Proc. of ICC'06, 2006, vol.9.

[9]. Nagarajan, R., Eylem Ekici1. An efficient and flexible MPLS signaling framework for mobile networks. // Wireless Networks, 2008, vol.14, no.6.

[10]. Sa-Ngiamsak. A Recovery Sheme for Qos Guaranteed Mobile IP Over MPLS Network. In: Proc. of Wireless Pervasive Computing, 2006, p.1-5.

[11]. Sahar, A., E. Shazely, A.Omayma, A. Mohsen, K.Shehatta. Enhancing MPLS Network Fault Recovery Using P-Cycle with QoS Protection. // IEEE CCC, 2007, IEEE Code: 1-4244-1430-X/07.

[12]. https://www.researchgate.net/post/Simulated_vs_Analytical_Results.

За контакти:
гл. асистент д-р инж. Венета П. Алексиева
катедра „Компютърни науки и технологии"
Технически университет Варна
E-mail: VAleksieva@tu-varna.bg

НИШКОВ СИМУЛАТОР ЗА ИЗЛЕДВАНЕ НА ПАРАМЕТРИТЕ ОТ КЕШ ПАМЕТИТЕ

Петя Кр. Димитрова, Мария Пл. Маринова, Владимир Д. Лазаров

Резюме: Представената тема описва кеш симулатора изграден с цел по-доброто разбиране на кеш архитектурата, откриване на предимствата и недостатъците при различното й конфигуриране. Този симулатор е изграден според основните принципи на кеш архитектурата и организацията. Той извежда като статистика качествените показатели на кеш от първо ниво с директна функция на съответствие. За реализирането на кеш симулатора е използван съвременен подход - език за описание на програмната част Си и използва GCC компилатор под Linux операционна система. Изграждането му е разделен на няколко етапа: създаване на среда за самостоятелно тестване, верификация на функционалността на кеш системата и симулация на работата на системата.
Ключови думи: симулатор, кеш памет, коефициент на непопадение, трейс файлове

Trace-Driven Similater for research to parameters of cache memories

Petya Kr. Dimitrova, Maria Pl. Marinova, Vladimir D. Lazarov

Abstract: This paper presents simulator of cache, which is created for studying architecture of cache systems. It could be in help of students and lector of Computer Architecture and Organizations. The results of this simulator are statistics of direct mapping function of caches. The simulator is write on C and used GNU compiler for LINUX. The creation of the simulator is divided on some parts: creating of a tool for making of independent tests, verification of the functionality of cache systems and simulation of cache's work.
Keywords: simulator, cache memoty, miss, memory traces

1. Увод

При съвременното развитие на компютърните системи един от основните компоненти, указващи голямо влияние върху производителността на компютърната система, е кеш паметта. Кешът е много критичен момент в работата на една компютърна система, което налага нуждата от доброто познаване на характеристиките и особеностите му. Важно е да се познават в дълбочина параметрите, които влияят на производителността му – размер на кеша, функции на съответствие, алгоритми на заместване, размер на блока, методи за запис и извличане. Добре проектираният и организиран кеш може да подобри поведението на цялата система. И обратно - не добре подбраната кеш конфигурация може да се отрази негативно върху производителността и бързината при работа на компютъра. Според областта, в която ще се използва, приложението, което ще намери и цялостната цена на компютърната система могат да се подберат кешове с различни характеристики.

2. Нишков симулатор на кеш паметите

Разработен е симулатор по студентски проект на Мерилендския университет [4]. Проектираният симулатор е функционален, *Trace-Driven Simulator (Нишков симулатор*). За работата с него е необходимо да се използва даннов файл. Тези файлове се наричат ***memory traces***, представляващи файлове, в които се съхраняват адресите от оперативната памет, към които ще се извърши обръщение в процеса на работа на симулатора**.** Производителността на

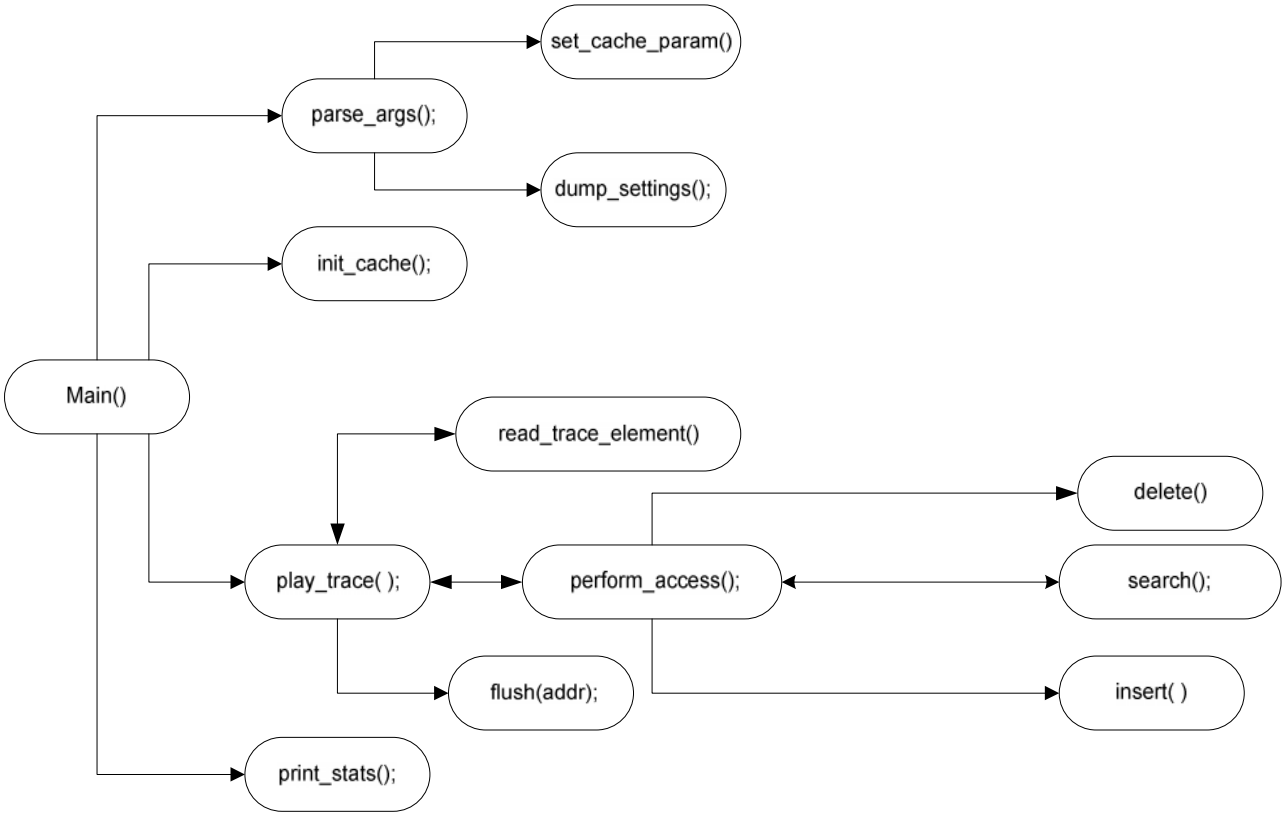
кеша зависи от програмата, която ще изпълнява процесорът. В случая трейс файлът е точно тази програма. За тестването на системата (кеш архитектурата) служи разработеният симулатор, предоставящ условия за изпълнение на трейс файлове (предоставени в заданието на проекта [6]) се получават различни изходни данни.

Използвани файлове:

- cc.trace (Си компилатор)
- tex.trace (обръщения на процесора към текстови документ)

Тези трейс файлове съдържат около 1 млн. обръщения към паметта. Те се различават по своето съдържание, тъй като за различните програми адресите на обръщение към паметта и достъпът до тях са различни.

Йерархичната структура на проектирания симулатор е показана на фигурата (фиг. 1).



Фиг. 1. Йерархичната структура на извикване на функциите на кеш симулатора

Функциите на симулатора могат да се разделят на три основни групи :

- **първа група** (*parse_args(),set_cache_param(), init_cache()*) са функции за настройка на елементите на кеш модела

  - функцията *parse_args()* взима ново въведената стойност от командния ред и я подава на функцията *set_cache_param()*, за да я присвои на съответната константа на кеш модела.

  - функцията *init_cache()* служи за инициализация т.е. начално установяване на елементите на кеш модела, определяне на тяхната разрядност и заделяне на памет.

- **втора група(***play_trace(), read_trace_element(), perform_access()***)** са функциите свързани с трейс файла:

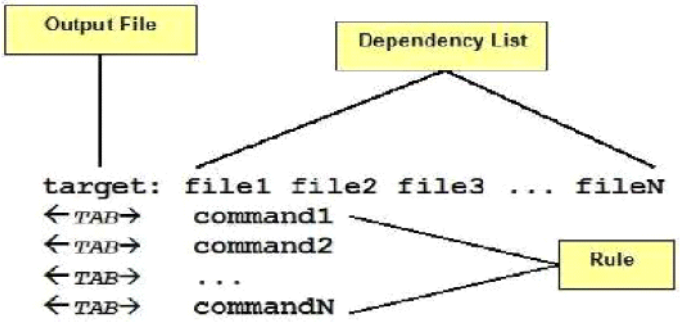
– отваряне на трейс файла за четене, четем от него типа на достъпа до паметта и адреса след това, в зависимост от прочетената стойност за достъп до паметта (т.е. четене, запис на данни или четене на инструкции)

- обработваме информацията *perform_access()*, намиране *search()*, ако е необходимо изтриваме *delete()* или вмъкваме данни *insert()*

- валидацията на данни *flush()*, която се използва при „обратен запис“ политика на запис т.е. проверка дали прочетените данни в кеша съвпадат с тези в оперативната памет.

• **трета група** (d*ump_settings() и print_stat()*) служат за извеждане на съобщение след края на симулацията.

Изграждането (построяването)на кеш симулатора се извършва в така наречения MAKEFILE. Чрез изпълнение на командата „**mak**e” в директорията, където се намира кодът на кеш симулатора, се получава изпълнимият изходен файл. *Makefile представлява:*



Фиг. 2. Основен синтаксис на *Makefile*

Използваните макроси са:
CC = gcc **CFLAGS** = -g
all: *sim*
sim: *main.o cache.o*
$(CC) -o sim main.o cache.o –lm
main.o: *main.c cache.h*
$(CC) $(CFLAGS) -c main.c
cache.o: *cache.c cache.h*
$(CC) $(CFLAGS) -c cache.c

В началото на файла са посочени макросите, които се използват съответно за **gcc** използваният компилатор и опцията **-g** за включване на дебъг символите.

Ще разгледаме файла отгоре надолу:

- Компилираме файла *cache.c* и получаваме обектен файл *cache.o*, като с опцията **–c** посочваме, че процесът се прекратява до получаване на обектния файл, а не до изпълнимия, т.е. предварителна обработка, компилация, асемблиране и без да се линква.

- Подобно на *cache.c* компилираме *main.c* и получаваме обектен файл *main.o*

- Компилираме двата обектни файла *main.o cache.o като чрез опцията –lm* добавяме опцията за аритметичната библиотека и получаваме изходен файл **sim**

Вече можем да изградим определен кеш модел като за целта зададем размерността на основните параметри от командния ред. При изпълнение на следната команда:

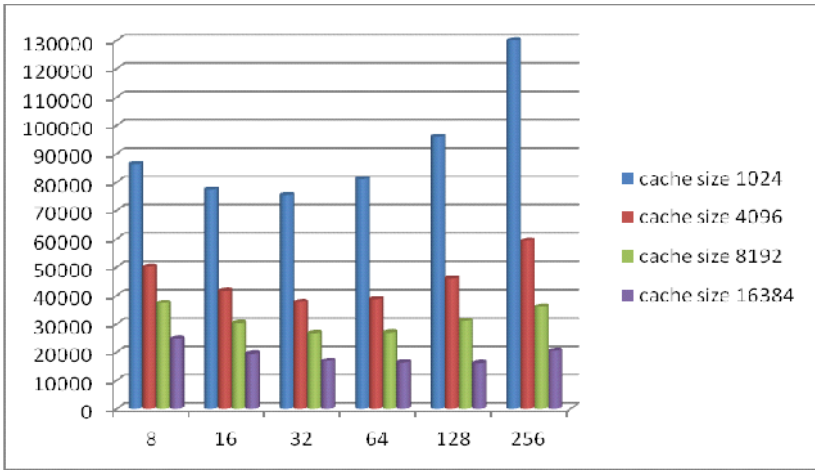
./sim -bs 64 -ds 16384 -is 16384 -a 1 -wb -wa ../trace/cc.trace

В крайна сметка след приключването на симулацията се извежда следното съобщение:

***** CACHE SETTINGS *****
Split I- D-cache
I-cache size: 16384
D-cache size: 16384
Associativity: 1
Block size: 64
Write policy: WRITE BACK
Allocation policy: WRITE ALLOCATE

***** CACHE STATISTICS *****
INSTRUCTIONS
accesses: 597309
misses: 135623
miss rate: 0.2271 (hit rate 0.7729)
replace: 135603
DATA
accesses: 235168
misses: 2673
miss rate: 0.0114 (hit rate 0.9886)
replace: 2417
TRAFFIC (in words)
demand fetch: 2212736
copies back: 2211152

Функцията *dump_settings()* отпечатва частта **** CACHE SETTINGS *** m.e.* конфигурацията на кеш модела, а *print_stats()* отпечатва втората част **CACHE STATISTICS **от съобщението за получените стойности на качествените параметри на съответно конфигурирания кеш модел. Едни от основните параметри, които се следят, са коефициентът на непопаденията и коефициентът на попаденията. Параметърът, които се извлича от получените статистически данни за оценка и анализ на производителността на кеш конфигурацията, е броят *на непопаденията в кеша.* На базата на получените данни се изграждат графики, чрез които по-качествено се илюстрира зависимостта на броя на непопаденията спрямо размера на кеша и размера на реда в кеша.



Фиг. 3. Графично представяне на получените резултати за непопадение(miss) в кеш за данни на различни конфигурации на размера на кеша и размера на блока в кеша за cc.trace

Получените стойности за изследвания параметър за непопадение (miss) или ненамерени данни при обръщение в кеша за инструкции показват, че при повишаване на размера на реда

(блока) на кеша от 8 на 16 байта ***miss*** параметъра намалява, следователно броят на попаденията (т.е. ***hit*** - намерени данни при обръщение в кеша) се увеличава, което се основава на принципа на близост в пространството. Това намаление на ***miss*** продължава до около 128 байта размер на блока в кеша, но при 256 и по нататък увеличение на рамера на блока коефициентът ***miss*** започва отново да се увеличава, тък като броят на блоковете, намиращи се в кеша, намалява. Това показва, че размерът на блока на кеша не може безкрайно да се увеличава, тъй като след даден момент това става безсмислено и нямаме полза от това. Използваните в съвременните компютри размер на реда в кеш от първо ниво е 64B.

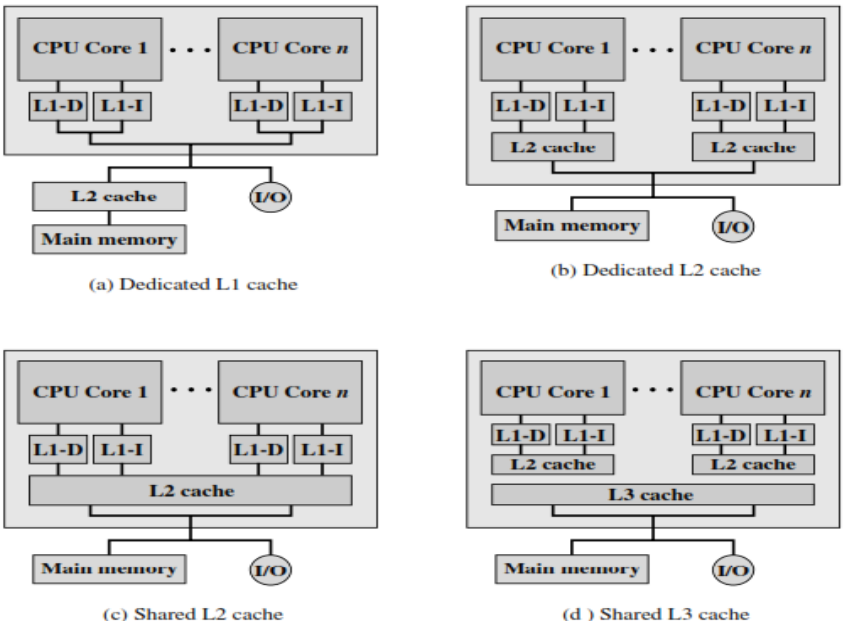
С повишаване на размера на кеша намалява коефициентът на непопадение ***miss*** и се увеличава коефициентът на попадение ***hit.*** Но размерът на кеш паметта не може безкрайно да се увеличава, тъй като времето на достъп до данните в кеша се увеличава с увеличаването на размера му. Това води до неефективно използване на кеш паметта и обезсмисля добавянето й.

3. Кешът при съвременните компютърни архитектури

Използваните съвременни ***Intel*** архитектури са ***NetBurst***, използваща ***Pentium 4*** процесор, ***Core, Nehalen***, използваща ***Intel core i7, Intel core i5, Intel core i3*** процесори. По долу в таблицата са описани кешовете на тези архитектури.

Таблица 1. Примерни параметри на съвременни Intel кеш архитектури

Ниво Видове	**L1(size)**	**L2(size)**	**L3(size)**
Intel core i7	32KB instruction and data cache	256 KB shared inst/mem cache	6MB shared inst/mem cache
Intel core i5	4x32 KB instruction and data cache	4x256 KB shared inst/mem cache	6MB shared inst/mem cache
Intel core i3	2x32 KB instruction and data cache	2x256 KB shared inst/mem cache	3MB shared inst/mem cache
Pentium 4	16 KB instruction and data cache	1024KB shared inst/mem cache	[1].



Фиг. 4. Илюстрира четирите основни организации на с-мите с множество ядра [2]

„Тази работа е подготвена в рамките на проект № ДЦВП 02/1, финансиран от ФНИ“.

Литература

[1]. ”Computer Organization and Design _The Hardware – Software Interface ” D. Patterson and J. Hennessey
[2]. William Stallings ”COMPUTER ORGANIZATION AND ARCHITECTURE *DESIGNING FOR PERFORMANCE „* EIGHTH EDITION , Prentice Hall , Upper Saddle River, NJ 07458
[3]. http://www.rapidtables.com/code/linux/gcc/gcc-g.htm
[4]. http://www.cs.umd.edu/

За контакти:
д-р Мария Пл. Маринова, ИИКТ-БАН
E-mail: itmpm@abv.bg

проф. Владимир Д. Лазаров, ИИКТ-БАН
E-mail: vladimir.lazarov@epu.bg

студент Петя Кр. Димитрова
ТУ - София, филиал Пловдив

НЕОТОРИЗИРАНО ПРОНИКВАНЕ В КОМПЮТЪРНА СИСТЕМА С АКТИВИРАНИ ЗАЩИТНА СТЕНА И АНТИВИРУСЕН СОФТУЕР

Петър Кр. Боянов, Жанета Н. Ташева

Резюме: В статията са представени резултати от осъществено неоторизирано проникване в операционната система Microsoft Windows XP SP3 с цел получаване на пълен достъп до ресурсите на избраната компютърната система. Атаката е реализирана успешно, въпреки че в дадения хост са активирани защитна стена и антивирусен софтуер. Благодарение на специално избрана платформа за злонамерено проникване и определен злонамерен експлойт за тази операционна система е получен пълен безжичен отдалечен достъп до ресурсите на този хост. Затова е необходимо да се използва по-сигурна и надеждна операционна система.
Ключови думи: Неоторизирано проникване, сканиране, уязвимости, злонамерен код.

An unauthorized penetration into computer system with activated firewall and antivirus software

Petar Kr. Boyanov, Zhaneta N. Tasheva

Abstract: In this article an unauthorized penetration into the operating system Microsoft Windows XP SP3 is accomplished. The aim is to gain full access to the resources of the chosen computer system. The attack is successfully made in spite of the activated firewall and antivirus software in the given host. Thanks to a specially chosen malicious penetration platform and specific malicious exploit for this operating system is gained full wireless remote access to all resources of this host. In order to prevent future malicious cyber-attacks it is necessary to be used more secure and reliable operating system.
Keywords: An Unauthorized penetration, scanning, vulnerabilities, malicious code.

1. Увод

През последните няколко години най-уязвимата операционна система за неоторизирано проникване е Windows на корпорацията Microsoft. Независимо че Microsoft ежедневно разработва различни критични ъпдейти за намаляване на уязвимостите в операционните си системи, много злонамерени потребители успяват да заобиколят различните защитните механизми и като крайна стъпка да проникнат в компютърната система на своята жертва. Най-големи загуби претърпяват потребителите, които използват електронно банкиране в глобалната мрежа Internet [1], [2], [3], [6], [7], [9], [10], [11].

Целта на тази статия е да се сканира операционната система Microsoft Windows XP SP3 за различни уязвимости и слабости и след това да се осъществи неоторизиран и незабелязан достъп в операционната система, с цел получаване на пълен достъп до ресурсите на компютърната система. Избрана е тази операционна система, защото все още много банки в Република България и български университети продължават да я използват, като поддръжката на Microsoft Windows XP SP3 ще продължи до април 2014 [10], [11].

Тази статия е структурирана в няколко раздела. В раздел 2.1 са представени и сравнени предишни научни разработки на различни злонамерени атаки. В раздел 2.2 е илюстрирано сканирането на безжичната локална мрежа за активни хостове. В раздел 2.3 е осъществено инжектиране на злонамерен код в операционната система на хоста-жертва. Резултатите от изследването са показани в раздел 3. Заключението и изводите от статията са представени в раздел 4.

2. Изложение

2.1. Предишни научни разработки

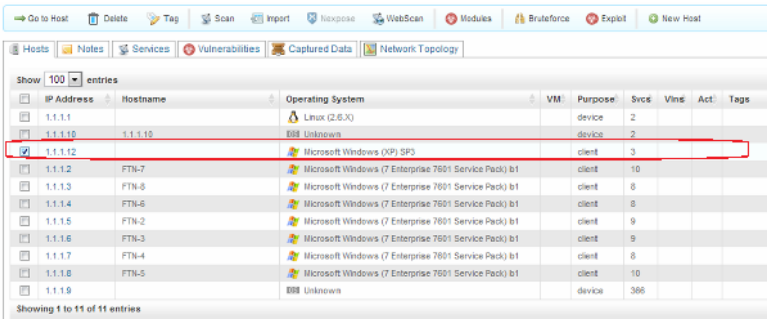
Различни видове злонамерени атаки с платформата Metasploit са разгледани и анализирани от Ramirez-Silva и Dacier [16]. Основната структура, както и отделните компоненти на Metasploit са обяснени и тествани от Moore [11], [12], [13]. Опити за осъществяване на неоторизиран достъп до ресурсите на отделни компютърни системи са осъществени от Henry [5] и Zovi [18]. Отделни начини за инжектиране на еклойти и допълните изпълними кодове са разгледани и показани от Fewer [4], Bradbury [1] и Dodd [3].

2.2. Сканиране на безжичната локална мрежа за активни хостове

Преди да бъде осъществена определена злонамерена кибер-атака, трябва да се обърне внимание на двата термина **експлойт** и **пейлоуд**. Терминът експлойт (exploit) представлява злонамерен софтуерен код, който използва уязвимост, грешка или бъг в компютърния хардуер или софтуер, някои от услугите (services) на операционната система, като по този начин предизвиква неочаквано и нежелано поведение на системата (атакуващият получава пълен достъп до системата) [17]. След успешното инжектиране на експлойта в системата се пренася като товар и същинският код (payload), който се изпълнява в хоста-жертва [2], [8], [9], [10], [14], [15]. За краткост по-долу вместо двата термина ще се използва злонамерен код.

Експериментът е направен в безжична локална мрежа WLAN (Wireless Local Area Network) от 15 хоста в компютърна лаборатория във Факултета по технически науки при Шуменски университет „Епископ Константин Преславски”. Всеки хост използва безжичен USB адаптер 150Mbps Wireless N USB Adapter TL-WN721N. В компютърната лаборатория се използва безжичен рутер 150Mbps Wireless N Router TL-WR741ND, на който е активиран протоколът DHCP (Dynamic Host Configuration Protocol) с цел автоматично раздаване на IP (Internet Protocol) адреси на всеки хост, който се опита да се свърже с рутера. Безжичната локална мрежа използва двадесет и четири битова мрежова маска и номерът на мрежата е 1.1.1.0, т.е. 1.1.1.0/24. Атакуващият хост се е свързал към безжичната мрежа и използва операционна система Microsoft Windows 7 Enterprise SP1 и платформата Metasploit Pro.

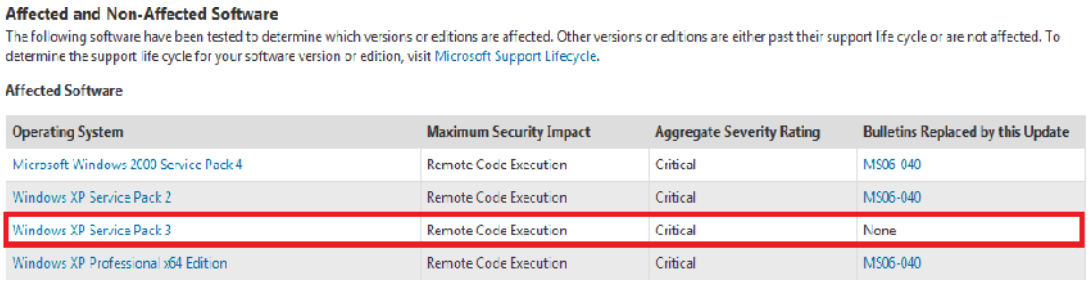
Първата задача е свързана със сканиране на безжичната локалната мрежа с цел установяване кои хостовете са в активно състояние. За осъществяването на тази задача е използван уеб-базираният потребителски интерфейс Access Metasploit Web UI. На фиг. 1 е показан резултатът от извършеното сканиране на локалната мрежа. От фиг. 1 се разбира, че са открити 11 хоста. Хостовете с IP адреси 1.1.1.10 и 1.1.1.9 са с неопределена операционна система и са също мрежови възли. Хостът с IP адрес 1.1.1.1 е всъщност рутерът в локалната мрежа и използва Linux базирана операционна система. Хостовете с IP адреси от 1.1.1.2 до 1.1.1.8 използват операционната система Windows 7 Enterprise SP1. Избраният хост-жертва е с IP адрес 1.1.1.12 и използва операционната система Microsoft Windows XP SP3. Фиг. 1 е направена в атакуващата машина.



Фиг. 1. Резултатът от сканирането на локалната мрежа и установяване, че хостът-жертва е с IP адрес 1.1.1.12 и използва операционната система Microsoft Windows XP SP3

2.3. Инжектиране на злонамерен код в операционната система на хоста-жертва

В хоста-жертва е активирана вградената в операционната система защитна стена (Firewall), и е стартиран антивирусният софтуер Avast Free Antivirus Version 8.0.1489. Важна стъпка за осъществяване на кибер-атаката е изборът на подходящ експлойт за дадената операционна система. Като най-подходящ е избран Microsoft Server Service Relative Path Stack Corruption - Server exploit от 28 октомври 2008 г. На фиг. 2 е показано, че за операционната система Microsoft XP SP 3 досега не е намерено противодействие срещу този злонамерен код и програмистите от Microsoft продължават да търсят някакво решение за премахване на тази критична слабост в операционната система Microsoft Windows XP SP3 [2], [10].



Affected and Non-Affected Software

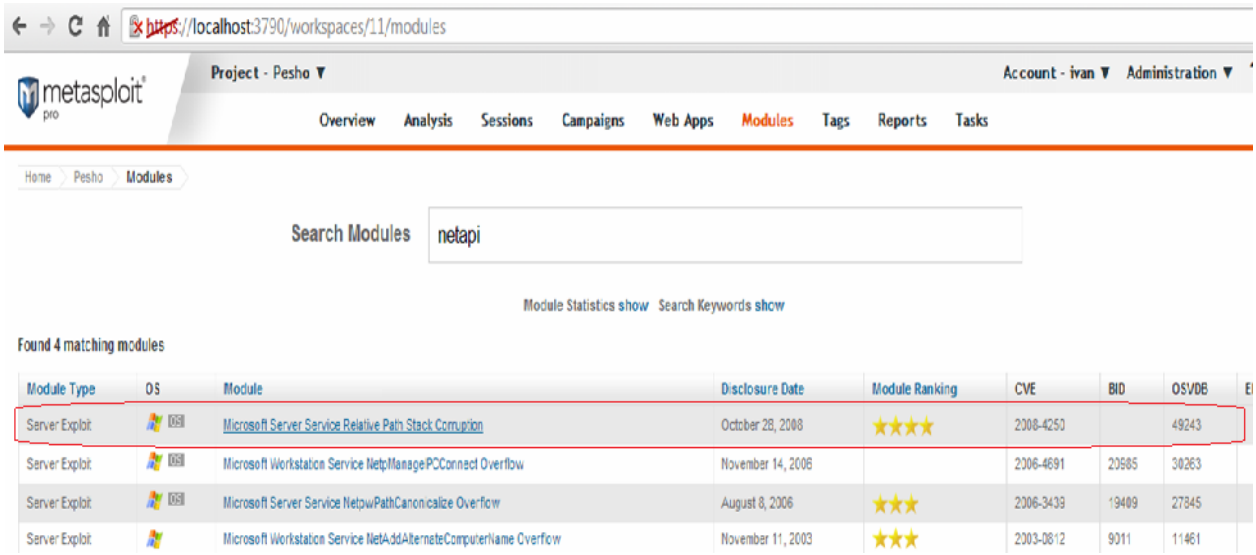
The following software have been tested to determine which versions or editions are affected. Other versions or editions are either past their support life cycle or are not affected. To determine the support life cycle for your software version or edition, visit Microsoft Support Lifecycle.

Affected Software

Operating System	Maximum Security Impact	Aggregate Severity Rating	Bulletins Replaced by this Update
Microsoft Windows 2000 Service Pack 4	Remote Code Execution	Critical	MS06-040
Windows XP Service Pack 2	Remote Code Execution	Critical	MS06-040
Windows XP Service Pack 3	Remote Code Execution	Critical	None
Windows XP Professional x64 Edition	Remote Code Execution	Critical	MS06-040

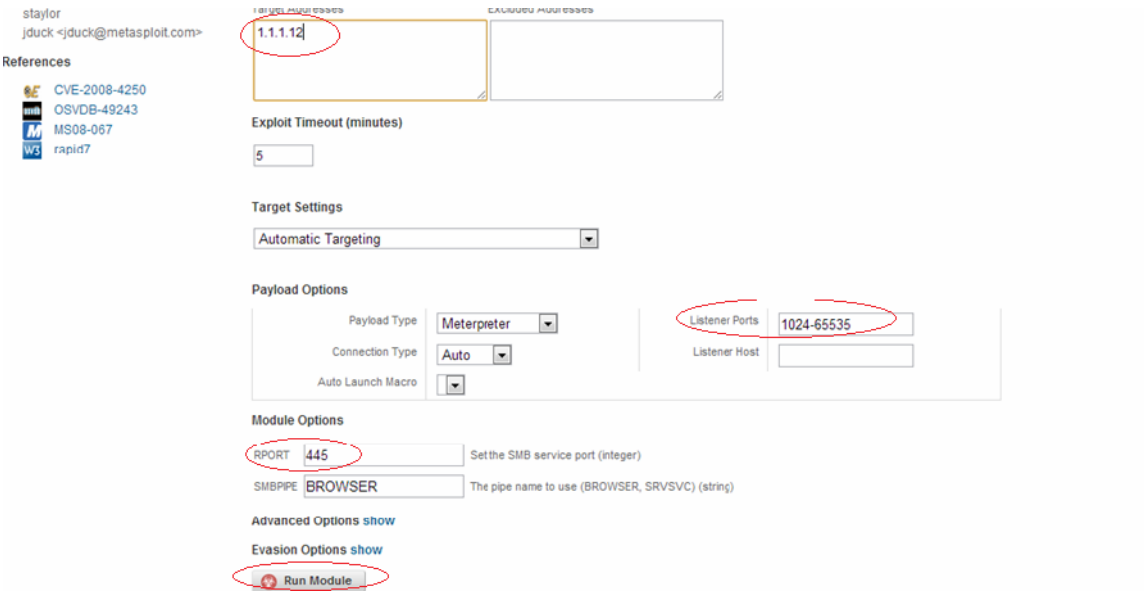
Фиг. 2. Корпорацията Microsoft и до ден днешен продължава да търси решение срещу този злонамерен код от октомври 2008 г.

Този злонамерен код позволява на кибер-престъпника отдалечено да изпълни случаен код чрез специално изработена RPC (Remote Procedure Call) заявка, която да предизвика препълване на стека със задачите в определената операционна система. На фиг. 3 е показано, че платформата Metasploit поддържа този злонамерен код и по този начин атакуващият може да го използва срещу хоста-жертва. От фиг. 3 се вижда, че избраният злонамерен код е с рейтинг 4 от максимално 5 в базата от данни на платформата Metasploit.



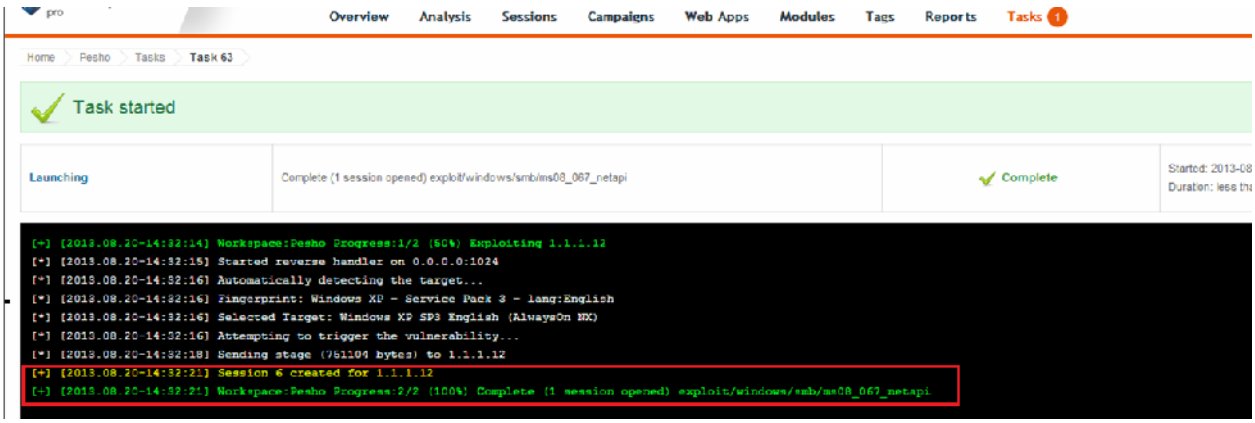
Фиг. 3. Microsoft Server Service Relative Path Stack Corruption

Настройките за инжектиране на злонамерения код Microsoft Server Service Relative Path Stack Corruption в операционната система Microsoft Windows XP SP3 са IP адрес - 1.1.1.12, порт 445, портове за слушане - от 1024 до 65535 и време за изчакване на действие на злонамерения код - 5 минути (фиг. 4).



Фиг. 4. Настройки за инжектиране на злонамерения код Microsoft Server Service Relative Path Stack Corruption

На фиг. 5 е показан успешно инжектираният злонамерен код в дадената операционна система с IP адрес 1.1.1.12. От тази фигура се разбира, че вече е създадена една сесия с този хост и по този начин хостът е компрометиран. Изобразеното на тази фигура е от екрана на машината на атакуващия. Времето за осъществяване на кибер-атаката е около 7 секунди, въпреки че има инсталирана антивирусна програма, която е настроена за дълбоко сканиране на цялата система.

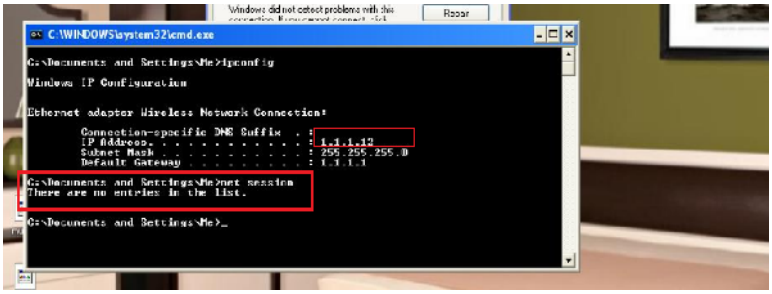


Фиг. 5. Успешно инжектиран злонамерен код и успешно създадената сесия на хоста-жертва

3. Резултати

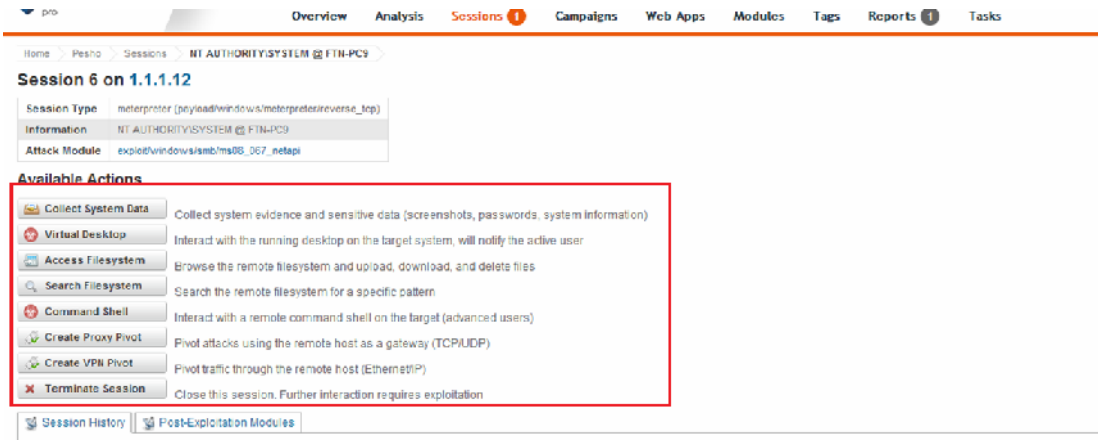
Реализираната кибер-атака използва известен още от 2008 г. експлойт, но е осъществена върху хост, където са приложени защитни механизми, като най-новите софтуерни защитни кодове (security updates), активирана вградена защитна стена и инсталирана антивирусна програма с най-новите дефиниции за вируси, но въпреки това жертвата е компрометирана за по-малко от 7 секунди.

Проверката в машината на жертвата за активни сесии (фиг. 6) не показва наличието на такива в операционната система, което от своя страна доказва, че атаката е протекла незабелязано.



Фиг. 6. Проверка за активни сесии на хоста-жертва

След успешно реализираната кибер-атака, атакуващият може да избере подходящи действия, които могат да бъдат извършени с жертвата. От фиг. 7 се разбира, че атакуващият може спокойно да събере важна информация за жертвата, като скрийншотове, пароли, системна информация, да сканира в реално време какво точно се случва на работния плот на жертвата, да търси определени файлове, да копира и да изтегли всеки файл от операционната система. Съществуват и възможности за използване на командния терминал (command shell). Всички възможни действия, които могат да бъдат извършени на хоста с IP адрес 1.1.1.12, са показани на фиг. 7.



Фиг. 7. Всички възможни действия, които могат да бъдат извършени на хоста-жертва

4. Заключение

Дори с активирани защитни механизми, хостът с операционна система Microsoft Windows XP SP3, е компрометиран и по този начин е получен достъп до неговите ресурси. Тази операционната система продължава все още да се използва в някои от отделите, особено финансовите, на фирми, университети и банки в България. Основната причина за използването на Windows XP е, че много от старите програми не са съвместими с по-новите операционни системи на Microsoft.

В резултат на проведения успешен експеримент в университетска лаборатория на ШУ „Епископ К. Преславски“ може да се предложат следните препоръки към администраторите по сигурността. Компютърните системи с операционна система Microsoft Windows XP SP3 да се използват за обработка и съхранение на не толкова важна и конфиденциална за учреждението информация. По възможност преминаване към по-нова операционна система, което ще повиши сигурността и надеждността на използваните безжични фирмени мрежи.

ЗАБЕЛЕЖКА: „Който копира, използва или осъществява достъп до компютърни данни в компютърна система без разрешение, когато се изисква такова, се наказва с глоба до три хиляди лева.” // Наказателен кодекс - Чл. 319а. (Нов - ДВ, бр. 92 от 2002 г.) (1) (Изм. - ДВ, бр. 38 от 2007 г.). Всичките експерименти са направени в специална лаборатория с

безжична локална мрежа от няколко хоста. Всичко илюстрирано и обяснено в статията е с научно-изследователска цел и ние не носим отговорност в случаи на злоупотреба с тях.

Благодарности

Тази статия е подкрепена по Проект BG051PO001-3.3.06-0003 "Изграждане и устойчиво развитие на докторанти, постдокторанти и млади учени в областта на природните, техническите и математическите науки". Проектът се осъществява с финансовата подкрепа на Оперативна програма „Развитие на човешките ресурси", съфинансирана от Европейския социален фонд на Европейския съюз.

Литература

[1]. Bradbury, D. Hands-on with Metasploit Express, Network Security, 2010(7), 7-11
[2]. CVE Details, The ultimate security vulnerability datasource, Available at http://www.cvedetails.com/cve/CVE-2008-4250/
[3]. Dodd, D. "Post exploitation using metasploit pivot & port forward.", 2011
[4]. Fewer, S. Reflective DLL injection, Harmony Security, Version, 1. 2008
[5]. Henry, P. A. Anti-Forensics, Secure computing, 2006
[6]. Irani, M. T., & Weippl, E. R. Automation of Post-exploitation. In Security Technology Springer Berlin Heidelberg, 2009, pp. 250-257
[7]. Jordan, C., ROYES, J., & WHYTE, J. Writing detection signatures. USENIX; login, 2005, 30(6), 55-61
[8]. Kessler, G. C. Anti-forensics and the digital investigator, In Australian Digital Forensics Conference, March 2007, p. 1
[9]. Masood, R., Um-e-Ghazia, U., & Anwar, Z., SWAM: Stuxnet Worm Analysis in Metasploit, In Frontiers of Information Technology (FIT), IEEE, December 2011, pp. 142-147
[10]. Microsoft Security Bulletin MS08-067 - Critical, Published: Thursday, October 23, 2008, Available at http://technet.microsoft.com/en-us/security/bulletin/ms08-067
[11]. Moore, H. D., "The metasploit project–open-source platform for developing, testing, and using exploit code", 2005
[12]. Moore, H. D., "Source list of attackers targeting metasploit.com", 2009
[13]. Moore, H. D., "Metasploitation." CanSecWest 2006, 2006
[14]. Liu, V. Metasploit antiforensics project.
URL: http://metasploit.org/research/projects/antiforensics, 2005
[15]. O'Gorman, J., Kearns, D., & Aharoni, M., Metasploit: The Penetration Tester's Guide, No Starch Press, 2011
[16]. Ramirez-Silva, E., & Dacier, M. (2007). Empirical study of the impact of metasploit-related attacks in 4 years of attack traces. In Advances in Computer Science–ASIAN Computer and Network Security, Springer Berlin Heidelberg, 2007, pp. 198-211
[17]. Security Street, Rapid7, Available at https://community.rapid7.com/docs/DOC-1567
[18]. Zovi, D. A. D., An encrypted payload protocol and target-side scripting engine, In Proceedings of the first USENIX workshop on Offensive Technologies, USENIX Association, August 2007, p. 8.

За контакти:
ас. инж. Петър Кр. Боянов, доц. д-р инж. Жанета Н. Ташева
катедра „Комуникационна и компютърна техника"
Шуменски университет „Епископ К. Преславски"
E-mail: peshoaikido@abv.bg
E-mail: zh.tasheva@mail.bg

EFFICIENCY ANALYSIS OF SOHO ROUTERS

Radosław Wróbel, Katarzyna Pentoś, Tomasz Długosz

Abstract: The article presents efficiency analysis of three popular models of SoHo (Small office Home) routers. Tests of efficiency based on bandwidth for UDP and TCP protocol in three topologies. Results are presented on charts and tables in manner, which enabled solid and clear comparison. Authors also presented glimpse into history of "SoHo networks" with underlined situation in Bulgaria and Poland.
Keywords: SoHo, LAN, router.

1. Introduction

First extensive network was devised at 1960s of the last century and named ARPANET (Advanced Research Projects Agency Network) which gave birth to computer network. That moment was a massive breaktrough in computer science and huge progress in this kind of networks – beginning from local area network (LAN) and finishing on global network which is Internet.

Nowadays internet has become one of the most popular form of media, for evidence number of Polish users of global network has risen significantly during the last decade (Table 1) [3].

Table 1. Internet use and frequency of use by indyviduals, 2011 (% of indyviduals) [3]

	Internet users and non-users			Frequency of use (on average)	
	Used internet within the last 3 months	Used internet within the last 12 months	Never used internet	Every day or almost every day	At least once a week (including daily use)
EU-27	71	73	24	56	68
BE	82	83	14	65	78
BG	48	51	46	37	46
CZ	70	73	24	41	63
DK	90	91	7	78	88
DE	81	83	16	63	77
EE	77	77	20	59	73
IE	75	77	21	55	71
EL	52	53	45	37	47
ES	67	69	29	48	62
FR	78	80	18	62	74
IT	54	57	39	49	51
CY	57	58	41	45	54
LV	70	72	27	53	66
LT	64	65	33	48	61
LU	90	91	8	76	86
HU	68	70	28	56	66
MT	68	69	30	55	66
NL	91	92	7	79	90
AT	79	80	18	59	76
PL	62	65	33	45	58
PT	55	58	41	42	51
RO	40	44	54	24	37
SI	67	69	29	54	64
SK	74	78	20	56	72
FI	89	89	9	76	86
SE	93	94	5	80	91
UK[1]	85	87	11	70	81
IS	95	95	4	88	94
NO	93	94	5	82	91
HR	58	60	39	44	55
TR	40	43	55	26	36

Such rapid development occured in a consequence of the community that crave information and has need to communicate with others. As a further reference we use various mutimedia – http sites (HyperText Transfer Protocol), electronic mail, internet calls VoIP (Voice over IP), Video conferencing, Video on Demand (VoD) (Table 2).

Computer network services evolution (mainly multimedia service) puts new demands on whole network and every single component of network. About details regarding analisis of use local computer network find out in [4]. In this thesis autors will focuse their efforts on analysing performance of routers used in home network LAN. Because of article size restrictions mechanizm, structure and used protocols in computer network will not be discussed. Further information enclosed in [5 - 7].

Table 2. Purposes of internet usage for Poland [3]

Internet activity	Year			
	2005	2006	2007	2008
Electronic mail use	24 %	27 %	32 %	38 %
Finding information about commodity and services	18 %	25 %	27 %	33 %
Chatting and discussion forums	15 %	18 %	26 %	31 %
Online journals	13 %	16 %	15 %	19 %
Television and radio online	6 %	10 %	13 %	18 %
Internet calls and videoconferencing	5 %	8 %	10 %	15 %
Online games, file downloads (games, music, films)	12 %	16 %	17 %	12 %

2. Router in small home computer network

Computers are becoming more and more popular devices in our homes (acquired in nearly 70 % households). In addition, acording to data more often we are owners of HD TV, mulitimedia player DVD, web camera or external hardware [3]. Devices mentioned above form our own amusement centre and possibilities are expanding when they're connected to the Internet. Such connection in small home network is done by router.

Routers used at home network LAN don't have to be very expensive. Essential features for router involve reliability, productivity and be not complicated with service (configuration usually updates through the internet). Basic features of the router [8, 9]:

- Internet connection.
- Quantity of comunication ports.
- Dynamic Host Configuration Protocol (DHCP).
- Network address Translation (NAT) and Port Address Transalation (PAT).
- Implemented firewall.
- Possiblity of forming Demilitarized Zone (DMZ).

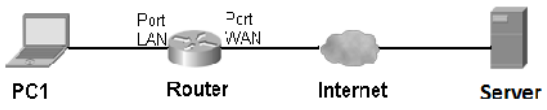
3. Methods of analysis

In the course of analysis we used three types of following routers:

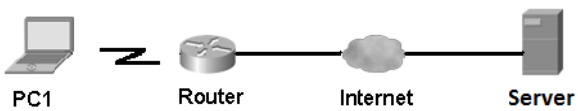
- Tp-link WR1043ND [10].
- D-link DI-524 [11].
- D-link DIR-300 [12].

Trials were conducted when devices connected to the router either via cable or wirelessly with a frequently changed topology variants (fig. 1) [13]:

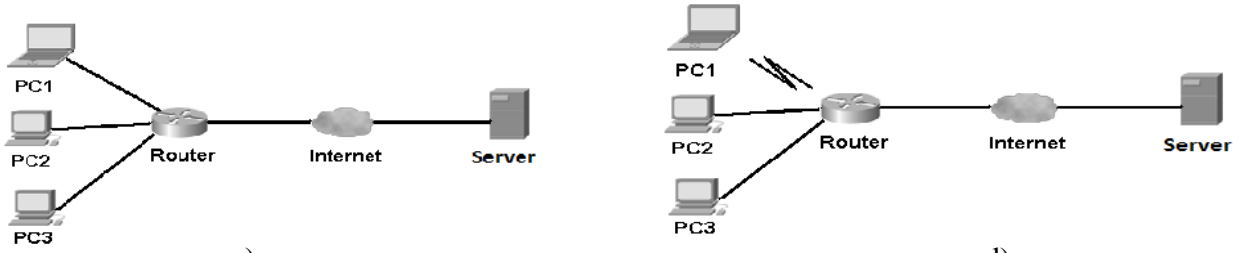
- Variant I – data transfer between host in local network PC1 connected with a cable with router and server that operate in Internet network (fig. 1a).
- Variant II – data transfer between host in local network PC1 connected wireless with router and server that operate in Internet network (fig. 1b).
- Variant III – local network is operated by three hosts. PC1 connects by cable with server that operates in Internet network. Hosts PC2 and PC3 generate additional motion in LAN network (fig. 1c).
- Variant IV – in local network occure 3 hosts. PC1 connected wireless with server working in internet network. Hosts PC2 and PC3 generate additional motion in LAN network (fig. 1d).



a)



b)



c) d)

Fig. 1. Different configuration types in home network chosen to trial:
a) variant I, b) variant II, c) variant III, d) variant IV

Among softwares, that were used in trials was used open source movement [14 – 18]:

- Netpref.
- Performance Test.
- Wireshark.
- WinAgent TFTP Server.
- Apache http Server.

In each trial devices configuration was the same, downloaded same files and ran same services.

4. Trial outcome

First group of results come from survey based on Transmission Control Protocol that was used to transfer data of 8129 bytes size. Second group of results refer to tests based on User Datagram Protocol (UDP), where transferred data of 1024 bytes size. Fig. 2 -6 (below) ilustrate outcome of conducted trials [13]. Fig. 2 shows capacity for network (variant I – fig. 1a) which works with protocol TCP. In this investigation router WR1043ND had the best performance either with sending or receiving information. Fig. 3 illustrates investigation outcome of wireless network LAN (variant II – fig. 1b) where capacity stays on the same level. In the case where for transfering data was used UDP protocol, difference of capacity fell significantly among each router (fig. 4 and 5). Fig. 6 shows results for variant III and variant IV network (fig. 1c and 1d), where additional motion in LAN network was generated by hosts PC2 and PC3 whereas host PC1 was connected to server HTTP. Tests were conducted relaying only on TCP. In this comparision router WR1043ND also had the best performance.

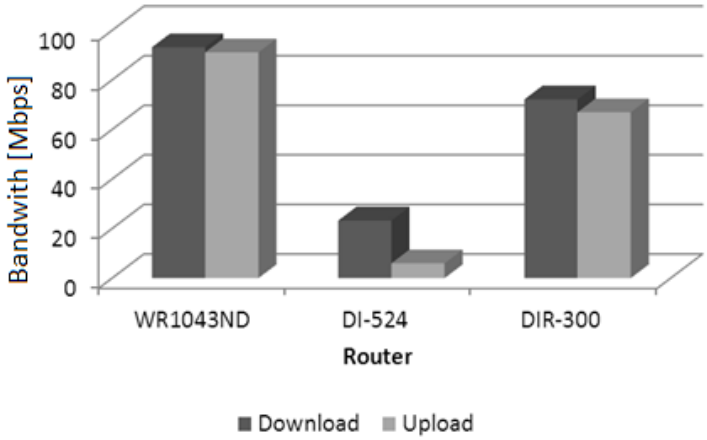


Fig. 2. Results of real bandwith of routers – variant I, TCP protocol

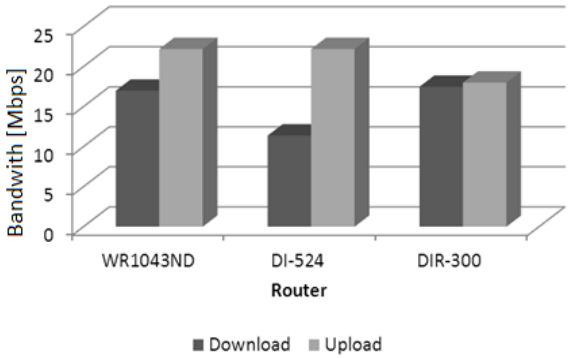


Fig. 3. Results of real bandwith of routers – variant II, TCP protocol

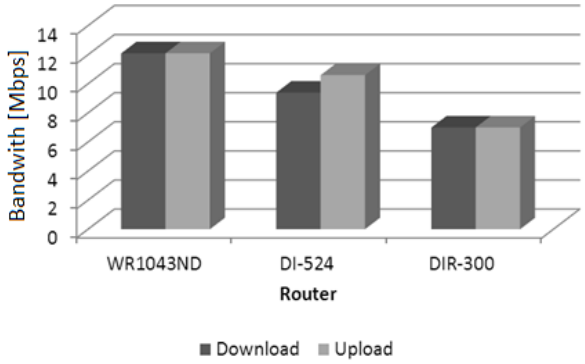


Fig. 4. Results of real bandwith of routers – variant I, UDP protocol

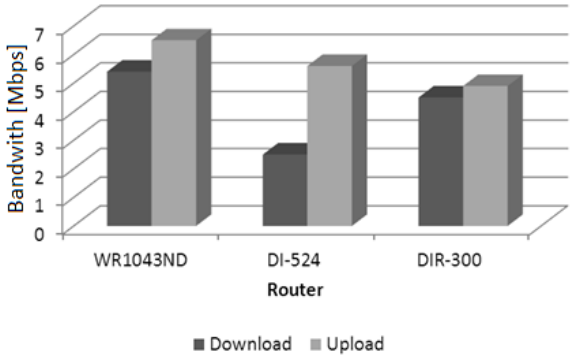


Fig. 5. Results of real bandwith of routers – variant II, TCP protocol

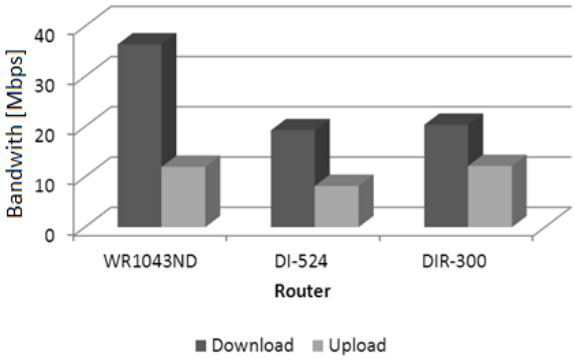


Fig. 6. Results of real bandwith of routers – variant III i IV, TCP protocol

Results figured 2 – 6 were repetitive. Each test was conducted few times. Furthermore few additional options were checked, for example measurements of transfer speed with different distances from router antena in wireless network or with coding swithed off what caused speed

improvement. Relation between routers were the same as presented above, therefore it will not be discussed here.

5. Summary

Thesis includes trials of particular routers for small office / home office use in computer networks. Data transfer speed is a single parameter that was examined. To maintain full image of tested devices additional features like functionality, structure, equipment and price should be tested, but this is not associated with the author's purpose. In the result of the trial the most productive device was WR1043ND router of Tp-link concern. The reason of the best performance of WR1043ND router is that this is the most recent device than the two others. Currently experience further improvements of routers efficiency. In fact whole electronic market is increasingly fast developing what we cheerfully wish for the future as well. So; not only price and extra facilities have influence on SoHo routers. The most important are topology and kind of data.

References

[1]. Lukasik S. Why The ARPANET Was Built, IEEE Annals of the History of Computing, Issue 99, March 2010, ss. 1-23.
[2]. Lynch D. C. Historical Perspective, rozdział w książce "Internet System Handbook", red. Lynch D. C., Rose M. T., Addison-Wesley, Reading, Mass, 1993.
[3]. Tlecenter Europe. http://www.telecentre-europe.org/?p=2038.
[4]. Długosz T., J. Huk. Usługi multimedialne w sieciach komputerowych, chapter in book „Biblioteka Teleinformatyczna, tom 5. Internet 2009", red. Bem D.J., Kasprzak A., Szymanowski M., Więckowski T., Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław 2010.
[5]. Tanenbaum A. S. Sieci komputerowe, Helion, Gliwice 2004.
[6]. Comer D. E. Sieci komputerowe i intersieci, WNT, Warszawa 2003.
[7]. Stevens W. R. Biblia TCP/IP, tom1. Protokoły, Addison Wesley, Warszawa 1998.
[8]. Milewski R. Routerem w sieć.
http://www.pcworld.pl/artykuly/31483_0_1/Routerem.w.Siec.html, May 2011.
[9]. Niedzielski D. Ewolucja routerów.
http://www.networld.pl/artykuly/349858/Ewolucja.routerow.html, May 2012. [10]. TP-Link website. http://www.tp-link.com.
[11]. D-Link ftp resources. ftp://ftp.dlink.pl. [12]. D-Link website. http://dlink.com.
[13]. Kozłowski M.: Analiza wydajności routerów do zastosowań domowych, PhD thesis, promotor dr inż. R. Wróbel, Wydział Informatyki, Wrocławska Wyższa Szkoła Informatyki Stosowanej, Wrocław 2011.
[14]. Netpperf homepage. http://www.netperf.org/netperf.
[15]. Passmark homepage. http://www.passmark.com/products/pt.htm. [16]. Wireshark homepage. http://www.wireshark.org.
[17]. Tftp-server homepage. http://www.tftp-server.com.
[18]. Apache homepage. http://httpd.apache.org./

For contacts:
Radosław Wróbel, PhD
Institute of Machine Design and Operation,
Wroclaw University of Technology,
Łukasiewicza 7/9, 50-371 Wrocław, Poland
E-mail: radoslaw.wrobel@pwr.wroc.pl

Katarzyna Pentoś, PhD
Institute of Agricultural Engineering,
The Faculty of Life Sciences and Technology,
Wroclaw University of Environmental and Life Sciences,
pl. Grunwaldzki 24A, 50-363 Wrocław, Poland
E-mail: katarzyna.pentos@up.wroc.pl
Tomasz Długosz, PhD
Telecommunications and Teleinformatics Department
Faculty of Electronics
Wroclaw University of Technology
Wybrzeże Wyspiańskiego 27,
50-370 Wrocław, Poland
E-mail: tomasz.dlugosz@pwr.wroc.pl

УПРАВЛЕНИЕ НА КОНФИГУРАЦИОННИ ФАЙЛОВЕ НА CISCO МРЕЖОВИ УСТРОЙСТВА ЧРЕЗ ПРОТОКОЛ SNMP

Делян Г. Генков

Резюме: Конфигурационните файлове са критични за мрежовите устройства на Cisco Systems. Без конфигурационен файл в повечето случаи устройствата не работят или не изпълняват някои важни функции. Съхранението на валидни конфигурационни файлове на устройствата е критично за мрежовата инфраструктура. Типичният начин за съхраняване на файловете се инициира от администратор чрез команди на самото устройство. Така за голяма мрежа трябва да се запазят конфигурациите ръчно за всяко устройство поотделно. Предложеният подход в тази разработка позволява централизирано управление на съхранението на конфигурациите на много устройства, използвайки протокол SNMP.
Ключови думи: Cisco, конфигурация, SNMP, TFTP, управление.

Managing configuration files of Cisco network devices using SNMP protocol

Delyan G. Genkov

Abstract: Configuration files are critical for Cisco systems networking devices. Without configuration these devices don't work at all or don't perform essential functions. Keeping copies of valid configuration files is critical for network infrastructure. Typical approach is initiated from an administrator typing commands onto device, so in a big network every device's configuration must be saved manually. Proposed approach in this papes allows a centralized management for saving the configurations of many devices, using SNMP protocol.
Keywords: Cisco, configuration SNMP, TFTP, management.

1. Увод

Устройствата на Cisco Systems са доста популярни в съвременните компютърни мрежи. Тяхната конфигурация представлява текстов файл с поредица от команди, съхранен в енергонезависима памет (NVRAM). При включване на електрозахранването те изчитат този файл, зареждат го в енергозависимата RAM памет, изпълняват командите от него и така добиват функционалността, която им е дадена от администратора. Без конфигурационен файл някои от устройствата (например повечето маршрутизатори) не правят нищо, а други работят със съвсем базова функционалност и не изпълняват важни функции.

Поддържането на актуални копия на конфигурационните файлове на устройствата е важна задача, особено в мрежи, където няма добре обучен администратор, който може да конфигурира бързо и правилно дадено устройство или в учебни лаборатории, където един обучаем може лесно да изтрие конфигурациите, направени от друг и така да повлияе негативно на обучението му.

Нормалният начин, предвиден от производителя за запазване на мрежовата конфигурация на устройствата предполага ръчно включване на администратор към всяко устройство, като чрез специална команда конфигурацията се запазва на външен TFTP сървър. Подходът е децентрализиран и предполага запазването на всяка конфигурация на устройство поотделно. При мрежа с много устройства процесът може да отнеме доста време и усилия.

Протоколът Simple Network Management Protocol (SNMP) е разработен за наблюдение и управление на параметри на мрежови устройства. Той е част от стандартния набор протоколи TCP/IP на Интернет и е дефиниран от IETF [1].

В типичните си приложения SNMP използва един или няколко компютъра с инсталиран специализиран софтуер, наречен мениджър, който наблюдава или управлява група мрежови устройства. На устройствата работи специализран софтуер, наречен агент, който отговаря на запитвания от мениджъра.

Агентите организират данните за наблюдение или управление в специални променливи. Променливите са организирани йерархично и тяхната организация, типове и описания са събрани в Management Information Base (MIB) – база данни за управление на устройството.

В момента съществуват три различни версии на протокола – версия 1, версия две с подверсии 2c и 2u и версия 3. В настоящата разработка се използва версия 2c. Дефинирани са седем типа съобщения:

- GetRequest: заявка от мениджъра към агента за получаване на стойност на променлива или списък стойности.
- SetRequest: заявка от мениджъра към агента за промяна на стойност на променлива или списък стойности.
- GetNextRequest: заявка от мениджъра към агента за откриване на достъпните променливи и стойностите им.
- GetBulkRequest: оптимизирана версия на GetNextRequest за много итерации.
- Response: отговор от агента, връщащ стойност или потвърждаващ заявка.
- Trap: асинхронно уведомяване от агента към мениджъра за настъпило събитие.
- InformRequest: потвърждение на Trap от мениджъра към агента [4].

Йерархичната база данни (MIB) може да съдържа стандартна за всички устройства информация, както и специфична за даден потребител. Информацията се състои от множество идентификатори на обекти (Object identifiers, OID), всеки от които описва един параметър или група параметри на устройството. Всеки идентификатор е поредица от числа и/или думи, разделени с точки, например за да се получи името, назначено на дадена мрежова система, трябва да се прочете идентификаторът .1.3.6.1.2.1.1.5.0, като поредицата числа описва пътя от корена на дървото на базата данни до конкретната стойност.

Връзката между мениджъра и агента се осъществява чрез въвеждане на Community string – дума, играеща роля на парола. Обикновено на устройствата могат да се задават две нива на пароли – RO, която се използва само за четене на параметрите и RW, която може да се използва и за запис на стойности. За реализация на Set заявки е необходимо да се разполага със стринга за четене и запис.

Протоколът SNMP се поддържа от много устройства на различни производители и е стандарт за мрежово наблюдение и управление.

2. Изложение

Базата данни MIB на устройствата на Cisco Systems съдържа идентификатор, наречен ciscoConfigCopyMIB, позволяващ манипулирането на конфигурационните файлове. Използваните в разработката идентификатори са:

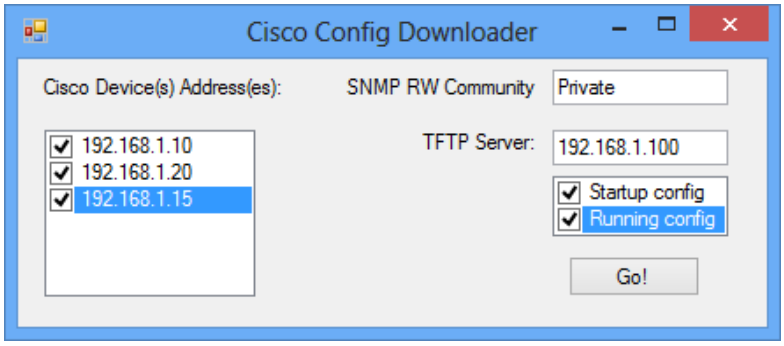
- Изпращане на SET заявка до 1.3.6.1.4.1.9.9.96.1.1.1.1.2.336 указва начина на предаване на файла: 1 означава протокол TFTP, 2 – FTP, 3- RCP, 4 – SCP и 5 – SFTP.
- Изпращане на SET заявка до 1.3.6.1.4.1.9.9.96.1.1.1.1.3.336 указва какво искаме да копираме: стойност 1 означава файл от мрежата, 2 – операционна система, 3 – запазената стартова цонфигурация, 4 – работната конфигурация в RAM, 5 – терминалния прозорец.

- Изпращане на SET заявка до 1.3.6.1.4.1.9.9.96.1.1.1.1.4 указва къде ще копираме, като стойностите са същите, като в горния идентификатор.
- Изпращане на SET заявка до 1.3.6.1.4.1.9.9.96.1.1.1.1.5 със стойност стринг, съдържащ IP адрес указва сървъра, на който ще се копира информацията.
- Изпращане на SET заявка до 1.3.6.1.4.1.9.9.96.1.1.1.1.6 със стойност стринг, съдържащ име на файл указва името на файла, който ще се запише на сървъра.
- Изпращане на SET заявка до 1.3.6.1.4.1.9.9.96.1.1.1.1.14 със стойност 1 стартира процедурата на копиране.

Като последна цифра на идентификаторите трябва да се добави уникално число за всеки набор заявки, което да съвпада за всяка от групата команди за даденото копиране. Вече използван номер не може да се използва отново в рамките на предварително дефинирания период за таймаут, който е 5 минути.

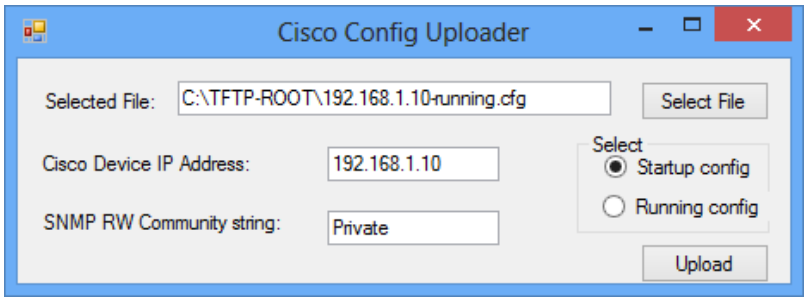
За настоящата разработка са написани две програми на Microsoft Visual Studio и езика C#. За реализация на функциите на протокола SNMP е използвана безплатната библиотека SNMP#Net.[3]

Първата програма е за съхранение на конфигурациите на мрежовите устройства. Главният екран на програмата е представен на фигура 1.



Фиг. 1. Програма за запазване на конфигурации

Програмата дава възможност да бъдат въведени IP адреси на устройствата, чиито конфигурации ще се четат. Последно въведените адреси се запомнят и се визуализират при следващо стартиране на програмата, а с маркерите могат да се активира и деактивира четенето на конфигурациите им. Трябва да се въведат Community string, който е настроен на устройството за запис на параметри, и адрес на TFTP сървъра, на който ще се записват конфигурационните файлове. Програмата дава възможност да се избере дали ще се записва началната (startup) конфигурация, записана в енергонезависима памет, работната (running) конфигурация, записана в енергозависимата памет или и двете. При натискане на бутона се осъществява комуникация с всяко от маркираните устройства в списъка, изпращат се към него описаните команди и стартовата конфигурация се копира на TFTP сървъра. Файловете се записват в директорията на TFTP сървъра с име, състоящо се от IP адреса, думата running или startup, указваща типа на конфигурацията и разширение cfg.



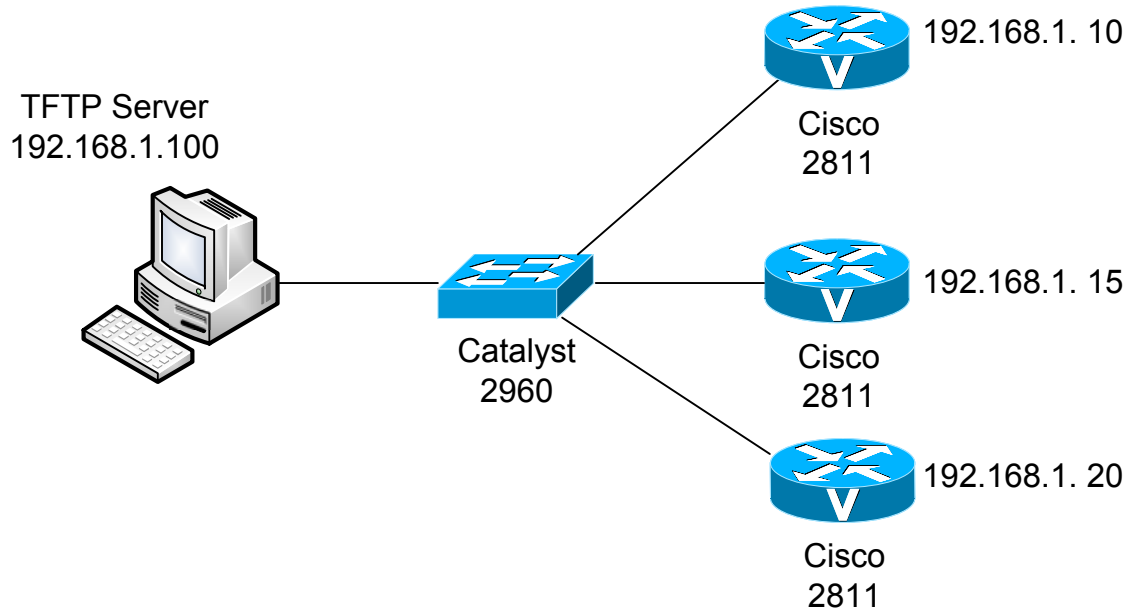
Фиг. 2. Програма за възстановяване на конфигурации

Екранът на втората програма е показан на фигура 2.

Програмата позволява на потребителя да избере файл с конфигурация, да напише IP адрес на устройството, където ще се изпраща конфигурацията и community string за достъп за четене и запис, да се избере дали ще се записва в стартовата или в работната конфигурация и с натискане на бутона да се изпрати конфигурацията към устройството.

3. Резултати

За проверка на работоспособността на предложения метод е изградена опитна постановка, показана на фигура 3. Тя се състои от три маршрутизатора, чиито конфигурации ще се четат, компютърна система, играеща ролята на SNMP мениджър, на който се стартират разработените програми и на TFTP сървър, където се съхраняват конфигурациите. Възможно е двете програми – SNMP мениджър и TFTP сървър да работят на различни компютърни системи. Като софтуер за TFTP сървър е използвана безплатната програма SolarWinds TFTP Server версия 10.9.0.25, маршрутизаторите са Cisco 2811, а операционната система на компютъра е Windows 8.



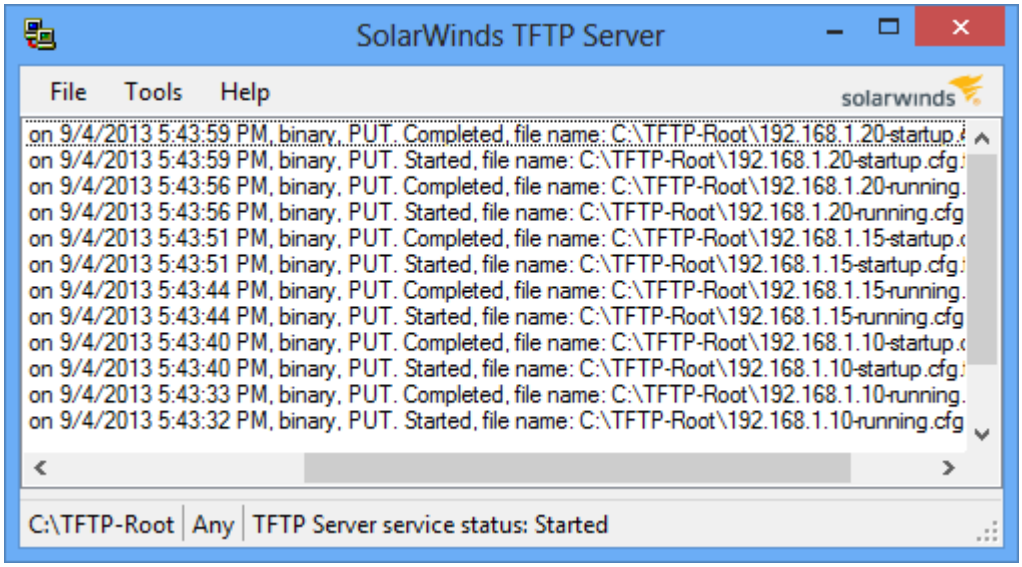
Фиг. 3. Схема на опитната постановка

За да сработи конфигурацията, на устройствата, чиито файлове ще четем и записваме трябва да се добави следният конфигурационен ред:

```
Router(config)# snmp-server community string Private rw
```

В примера текстът Private съответства на community string, който трябва да бъде изписан и в съответните екрани на приложенията. Той трябва да съвпада, за да може да се осъществи успешна връзка. Препоръчително е при конкретни реализации от съображения за сигурност да се използва различен текст, тъй като стрингът Private е общоизвестен и често използван по подразбиране в много мрежови устройства.

Резултатът от работата на програмите е представен чрез екрана на TFTP сървъра на фигура 4. Всички конфигурации са изтеглени успешно. Тестовете с програмата за зареждане на конфигурация в устройството също бяха успешни.



Фиг. 4. Успешно приемане на файловете

4. Заключение

В настоящата разработка е представен подход за централизирано управление на конфигурационните файлове на Cisco мрежови устройства. Разработени са софтуерни приложения за четене на конрфигурационните файлове от много устройства наведнъж и за зареждане на избран конфигурационен файл в устройство. Приложенията са тествани в лабораторията на Cisco мрежовата академия към технически университет – Габрово и са доста полезни в учебна среда, където често се налага да се зареждат различни конфигурационни файлове за симулацията на различни лабораторни упражнения.

Бъдещите планове по приложението включват интегриране на функционалност на TFTP сървър в самото приложение, за да се избегне използването на външна програма, добавяне на функционалност за сценарии при актуализиране на конфигурациите, което ще помогне за бързо реализиране на тестови постановки и лабораторни упражнения и реализация на комуникация по сигурен криптиран протокол – SFTP.

Литература

[1]. Harrington, D., R. Presuhn, B. Wijnen. An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks. IETF, RFC 3411, 2002.
[2]. Semperboni, F., “How to save configurations using SNMP”, CiscoZine, 22.08.2013
[3]. SNMP#Net Library, http://www.snmpsharpnet.com/, дата на използване 10.08.2013.
[4]. Wikipedia. Simple Network Management Protocol. http://en.wikipedia.org/wiki/Simple_Network_Management_Protocol, дата на използване 07.07.2013.

За контакти:
гл. ас. д-р Делян Генков
катедра „Компютърни системи и технологии”
Технически университет - Габрово
E-mail: dgenkov@tugab.bg

A COMPARISON OF COMPETITIVE NEURAL NETWORKS AND HIERARCHICAL CLUSTERING IN HONEYS CLASSIFICATION BASED ON CHEMICAL PARAMETERS

Katarzyna Pentoś, Deta Łuczycka, Radosław Wróbel

Abstract: Clustering is a fundamental data analysis method. Cluster analysis is used to group data in many fields. It's developed here as a tool to study chemical parameters of Polish honeys. In order to apply this technique, honey chemical features are shown by a vector with 6 coordinates representing: glucose content/ fructose content ratio, proline content, water activity of liquid honey, pH, diastase, HMF content. For clustering honey samples according to chemical parameters, Competitive Neural Network and hierarchical clustering were used. Results given by both methods used in this work are similar and demonstrated that chemical parameters characterize honeys in accordance with the honey type.
Keywords: cluster analysis, Artifitial Neural Networks, hierarchical clustering

1. Introduction

Clustering is a fundamental data analysis method. Cluster analysis (CA) refers to a set of analytic procedures that reduce multidimentional complex data into smaller groups (clusters). The group of CA methods is integrated into many popular statistical software packages therefore it is widely used in many research fields like medicine, biology, food processing technology, agricultural engineering etc. [1-5]. Except classic CA methods like hierarchical clustering or k-means clustering, other methods, like Artificial Neural Networks are used. The choice of clustering method depends on the nature of data and different methods can provide different results.

Bee honey is a foodstuff produced by bees (*Apis Mellifera*) with complex chemical composition. The major components of honey (about 80%) are sugars, mostly fructose and glucose. Additional compounds are carbohydrates, amino acids and proteins, volatile compounds, minerals and vitamins [6]. Honey composition depends not only on substrates but also on geographical origin, season and environment. In the European Union the composition and manufacture of honey are regulated by Community Directive 74/409/EEC [7].

Honey is used for human consumption for its medical and therapeutic effects. Its immense health benefits as antibiotic, anticancer, antioxidative, antibacterial and preserving activity as well as positive effects on the immune system are well known [8,9]. Lately, the demand for natural honey has increased [10]. In result, it has become an expensive product. Therefore, more and more producers adulterate honey by feeding bees with sugar (cane sugar), adding inexpensive sweeteners such as corn syrups, high fructose corn syrups, invert syrups or high fructose inulin syrups. As a result, health benefits of honey are reduced or eliminated [11-13]. Also overheating can decrease the quality of honey. Heating up is necessary during packaging process for honey liquefying. However, heating up destroys many important and valuable honey components [14,15].

Analitycal methods make accurate determination of honey chemical composition possible. The influence of adulteration methods on particular components reduction is the subject of many studies. It's not possible to indicate only one component, which can be used for detection of specific adulteration method. Especially that the content of various chemical components in honeys can be different depending on honey type, geographical origin and other factors, what doesn't decrease medical and therapeutic effects of honey. The full chemical analysis, including pollen analysis, is very expensive and requires specific and very expensive laboratory equipment. Therefore,

limitation on the number of chemical parameters which are used for honey types differentiation is crucial.

In this study two mathematical methods of honeys clustering according to their chemical features were used. Clustering process was performed using Competitive Neural Network and hierarchical clustering. The aim of study was to compare results given by these two methods in honeys classification based on chemical parameters and to determine the nature of the similarity of samples classified into the same cluster by each method.

2. Materials and methods

2.1. Honey samples

In this work a total of 50 honey samples were used. Honey samples were directly obtained in 2011 from private producers. Regarding the type of honey, thirty nine samples were nectar honeys, four samples were nectar-honeydew honeys and seven samples were honeydew honeys. In nectar honeys group, acacia, rape, phacelia, goldenrod, buckwheat, heather, willow and many multifloral honeys have been investigated. In honeydew honeys group, both, conifers and deciduous have been tested. During sample collecting, honey type variety was very important.

For all samples, honey type was verified on the basis of pollen analysis accomplished in accredited laboratory. Classification of honey sample to nectar-honeydew or honeydew honey group was based additionally on the value of 20% honey aqueous solution conductivity.

For all honey samples the following chemical parameters were measured: glucose content/ fructose content ratio [%], proline content [mg/100g], water activity of liquid honey [-], pH [-], diastase [-], HMF (*Hydroxymethylfurfural*) content [mg/kg].

2.2. Competitive Neural Networks

Competitive Neural Network (CNN) is a single layer neural network. All neurons in output layer (competitive layer) are connected with all neurons in input layer. Each neuron in competitive layer is associated with a reference vector (the weights vector). The data vectors are presented to the input. The input layer only transmits informations towards competitive layer. In the applied science there are many different algorithms which can be used for CNN training. The very often used is Kohonen algorithm. In this algorithm, the distance between input vector and the weight vectors is calculated. The Euclidian distance is often used (Eq.1).

$$d(x, w_i) = \|x - w_i\| = \sqrt{\sum_{j=1}^{N} (x_j - w_{ij})^2} \tag{1}$$

where x is input vector, w_i is weights vector of i neuron in competitive layer.

Because of the distance calculation, input vector elements and weights should be normalized to the same range, usually <0 – 1>. The neuron, which weights vector is closest to current input vector, becames the winner (Eq.2). The winner's output is 1 and outputs of all other neurons are 0.

$$d(x, w_w) = \min_{1 \le i \le n} d(x, w_i) \tag{2}$$

where $d(x,w)$ is the distance between input vector and weights vector, n is amount of neurons.
The weights of the winning neuron are adjusted with the Kohonen learning rule (Eq. 3).

$$w_i(k+1) = w_i(k) + \eta(k)[x - w_i(k)] \tag{3}$$

where x is input vector, w_i is weights vector of i neuron in competitive layer, η is the learning rate.

The neuron which weights vector was closest to the input vector is updated to be closer to input vector. Therefore, the winner is more likely to „recognize" input vectors similar to presented input vector and less likely to „recognize" input vectors different from presented input vector. After learning process, neurons in competitive layer can „recognize" groups of input data. The competitive network learns to categorize the input vectors from a learning set. Competitive networks have one important limitation. The distance between weights vector of some neurons at the beginning of learning process and input vector presented during net training may have high value. These neurons don't have oportunity to became winner and adjust weights. These neurons are called dead neurons and never perform a useful function [16].

2.3. Cluster analysis (CA)

The aim of clustering is to devide a set of objects into groups (clusters) that are similar in some given sense. Objects in a given group or cluster have a high degree of similarity and those which are in other clusters are different using the same criterion. Cluster analysis (segmentation analysis, taxonomy analysis), first used by Tryon in 1939 [17], is the group of clustering algorythms and methods. CA can be used only for detecting clusters in a set of data without explanation the nature of similarity. It is crucial to define a way to quantify the similarity of two objects (how close or how far two objects are from each other). The most intuitive metric for calculating distance between two points in Euclidean space is Euclidean distance (Eq.4).

$$d(x,y) = \sqrt{\sum_i (x_i - y_i)^2} \tag{4}$$

The other typical metrics used for distance calculation are as follows:
Squared Euclidean distance (Eq.5) gives the greater weight on objects that are further apart.

$$d(x,y) = \sum_i (x_i - y_i)^2 \tag{5}$$

City-block (Manhattan) distance (Eq.6) is the average difference across dimensions.

$$d(x,y) = \sum_i |x_i - y_i| \tag{6}$$

Chebychev distance (Eq.7)

$$d(x,y) = Max|x_i - y_i| \tag{7}$$

When there is the difference in values range (i.e. one variable range is from 0 to 10 and the second is from 100 to 10000), distance calculation based on raw data can cause the significant effect of one variable on the distance value. In this case, data standardization is recommended.

The two main clustering algorythms are hierarchical clustering and K-means clustering.

In hierarchical clustering the data are grouped into clusters using an iterative algorithm. The aim of hierarchical clustering is joining objects together into successively larger clusters on the basis of the metric of similarity. The results of hierarchical clustering are usually represented by hierarchical tree (dendrogram).

At the beginning of the algorythm, each object represents a class by itself. Then, in many steps, more and more objects are linked together making larger and larger clusters. In each iteration, clusters consist of increasingly dissimilar objects. Finally all elements are joined together. The five most popular linkage rules can be used for hierarchical clustering. *Single linkage* (nearest neighbor) is based on calculation the distance between two clusters as the distance of the two closest objects in these clusters. *Complete linkage* (furthest neighbor) is based on determining the distance between two clusters using the greatest distance between any two objects in these clusters. *Centroid linkage* uses the distance between the centroids of the two clusters, when the *centroid* of a cluster is the average point in the multidimensional space defined by the dimensions. *Average linkage* uses the

average distance between all pairs of objects in the two different clusters. *Ward linkage* uses variability within each cluster to calculate the similarity between clusters.

In K-means clustering the number of clusters is given a priori. The aim of the algorithm is to produce *k* clusters in which the elements within each cluster are as close to each other as possible and as far from the other elements in other clusters as possible. The algorythm starts with *k* random clusters, and then move objects between those clusters in order to minimize variability within clusters and maximize variability between clusters [18].

3. Results

For this study Competitive Neural Network and hierarchical clustering were used. Experimental data values were in a broad range, therefore data vectors were scaled into a new range of <0.1–1>.

3.1. The results of clustering using Competitive Neural Network

The simulations were executed using *Neural Network Toolbox – Matlab* environment. The number of input nodes was adequate to the number of parameters describing honey samples. The number of neurons in output layer and the number of training epochs were empirically determined.

For this work, neural network with seven input nodes and fifteen neurons in output layer was defined. The number of training epochs was determined as 800. Table 1 shows the results of honeys clustering according to honey type.

Table 1. The results of honeys clustering based on six chemical parameters using Competitive Neural Network according to honey type.

	Cluster 1	**Cluster 2**	**Cluster 3**	**Cluster 4**	**Cluster 5**
Honey type	acacia 1	honeydew 1	honeydew 3	buckwheat 1	goldenrod 1
	acacia 2	honeydew 1	honeydew 4	buckwheat 2	goldenrod 2
	acacia 3	multiflower 2	honeydew 5	buckwheat 3	goldenrod 3
	multiflower 1		honeydew 6	heather 1	phacelia 1
	multiflower 4		honeydew 7	heather 2	phacelia 3
	multiflower 9		nectar-honeydew 1	multiflower 3	rape 8
	nectar 1			multiflower 5	
	nectar 3			multiflower 6	
	nectar-honeydew 4			multiflower 7	
	rape 1			multiflower 8	
	rape 10			nectar 2	
	rape 2			nectar 4	
	rape 4			nectar-honeydew 2	
	rape 5			nectar-honeydew 3	
	rape 6			phacelia 2	
	rape 7			rape 3	
	rape 7				
	rape 9				
	willow 1				

All fifty honey samples were classified into five groups. Cluster 1 consists of 19 honeys, mostly acacia, multiflower and rape honeys. Honeydew honeys were devided into two groups (cluster 2 and cluster 3). Buckwheat, heather, five multiflower and two nectar-honeydew honeys were classified into cluster 4 and goldenrod with phacelia honeys were classified into cluster 5.

3.2. The results of clustering using hierarchical clustering

The hierarchical clustering simulation was executed using *Statistica v. 10* environment. For simulation, normalized data set was used. As distance metric Euclidean distance was chosen. Aglomeration process was performed according to Single linkage method. Fig.1. shows the cluster dendrogram obtained for 50 honey samples according to 6 chemical parameters characterising various types of honey. Honey samples were classified into four groups according to the similarity coefficient of 0.32. The first cluster includes mostly rape nectar honeys. Into the second cluster honeydew and nectar- honeydew honeys were classified. Groups 3 and 4 are similar to groups 4 and 5 from Competitive Neural Network clustering and include multifloral with buckwheat honeys and goldenrod with phacelia honeys, respectively.

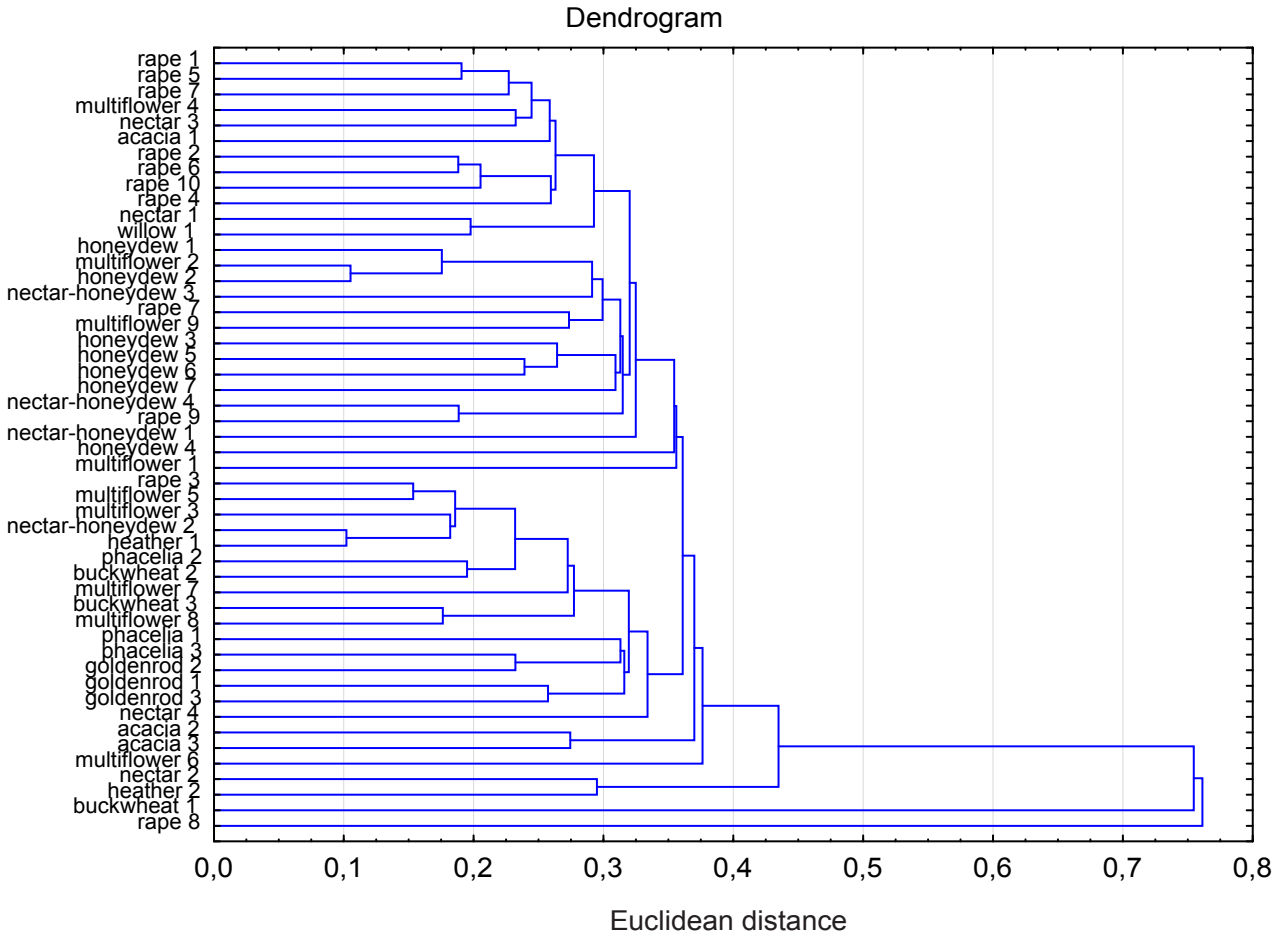


Fig. 1. Cluster dendrogram generated for fifty honey samples according to six chemical parameters

The results received using hierarchical clustering are generally consistent with the results recived using Competitive Neural Network. Only eleven samples were not classified into any cluster. That situation was observed in the case of samples described in table 1 as clasified into cluster 1 (3 samples), cluster 3 (2 samples), cluster 4 (5 samples) and cluster 5 (1 sample).

4. Conclusions

The results of honeys clustering show that chemical parameters characterize honeys in accordance with the honey type determined on the basis of pollen analysis. Classification of different honey types into the same cluster is the result of their chemical features similarity. Classification of given honey type sample into the same cluster with the samples of different honey types can be the result of the method of determining honey type based on pollen analysis. The honey type is determined according to predominant pollen. However, content of other pollen types can cause the chemical similarity of given sample to different honey type.

The results received using Competitive Neural Network and hierarchical clustering are similar. In case of hierarchical clustering some groups were connected into one cluster and eleven samples were not classified into any cluster.

Acknowledgements

This study was performed as part of project supported by Polish National Science Centre (grant no. N N313 766640).

References

[1]. Zhang Ch., J. Li, Z Zhu., Y. Zhang, L. Zhao, C. Wang. Cluster Analysis on Japonica Rice (*Oryza sativa* L.) with Good Eating Quality Based on SSR Markers and Phenotypic Traits, Rice Science, 2010, 17(2): 111–121.

[2]. Arciola C. R., D. Campoccia, L. Baldassarri, V. Pirini, J. Huebner, L. Montanaro. The role of Enterococcus faecalis in orthopaedic peri-implant infections demonstrated by automated ribotyping and cluster analysis, Biomaterials, 2007, 28, 3987–3995.

[3]. Andre's-Agusti'n J., F. Gonza´lez-Andre´s, R. Nieto-A´ngel, A.F. Barrientos-Priego. Morphometry of the organs of cherimoya (Annona cherimola Mill.) and analysis of fruit parameters for the characterization of cultivars, and Mexican germplasm selections, Scientia Horticulturae, 2006, 107, 337–346.

[4]. Medina W., O. Skurtys, J.M. Aguilera. Study on image analysis application for identification Quinoa seeds (Chenopodium quinoa Willd) geographical provenance, LWT - Food Science and Technology, 2010, 43, 238–246.

[5]. Rahman A. F., J. A. Gamon. Detectingbiophysical properties of a semi-arid grassland and distinguishing burned from unburned areas with hyperspectral reflectance, Journal of Arid Environments, 2004, 58, 597–610.

[6]. Borawska J., W. Bednarski, J. Gołębiewska. Charakterystyka sacharydów miodu oraz możliwości zastosowania Bifidobacterium do modyfikacji ich składu i właściwości, Żywność. Nauka. Technologia. Jakość, 2011, 3 (76), 29 – 39.

[7]. OJEC L 221, 12.8.1974

[8]. Blasa M., M. Candiracci, A. Accorsi, M.P. Piacentini, M.C. Albertini, E. Piatti. Raw Millefiori honey is packed full of antioxidants, Food Chem., 2006, 97, 217-222.

[9]. Kędzia B., E. Hołderna-Kędzia. Miód. Skład i właściwości biologiczne, Przeds. Wyd. Rzeczpospolita SA, Warszawa 2008.

[10]. Arvanitoyannis I. S., C. Chalhoub , P. Gotsiou , N. Lydakis-Simantiris & P. Kefalas. Novel Quality Control Methods in Conjunction with Chemometrics (Multivariate Analysis) for Detecting Honey Authenticity, Critical Reviews in Food Science and Nutrition, 2005, 45:3, 193-203.

[11]. Tuszyński T., M. Czernicka. Zafałszowanie żywności i napojów oraz metody ich wykrywania, Laboratorium Przegląd Ogólnopolski, 7-8/2008

[12]. Śmiechowska M. Selected problems of authentication and traceability of organic food, Journal of Research and Applications in Agricultural Engineering, 2007, Vol. 52(4) (in Polish).

[13]. Rios-Corripio M.A., M. Rojas-López, R. Delgado-Macuil. Analysis of adulteration in honey with standard sugar solutions and syrups using attenuated total reflectance-Fourier transform infrared spectroscopy and multivariate methods, CyTA - Journal of Food, 2012, 10:2, 119-122.
[14]. Lempka A. (red.). Towaroznawstwo produktów spożywczych, wydanie II, Państwowe Wydawnictwo Ekonomiczne Warszawa 1975
[15]. White J.W. The role of HMF and diastase assays in honey quality evaluation, Bee World, 1994, 75(3), 104-117.
[16]. Ossowski S. Sieci neuronowe do przetwarzania informacji, Oficyna Wydawnicza Politechniki Warszawskiej, Warszawa 2000
[17]. Tryon R. C. Cluster Analysis, Edwards Brothers, Ann Arbor, MI, 1939
[18]. Dobosz M. Wspomagana komputerowo statystyczna analiza wyników badań wydanie drugie uaktualnione, wyd Exit 2004

For contacts:
Katarzyna Pentoś, PhD
Institute of Agricultural Engineering,
The Faculty of Life Sciences and Technology,
Wroclaw University of Environmental and Life Sciences,
E-mail: katarzyna.pentos@up.wroc.pl

Deta Łuczycka, PhD
Institute of Agricultural Engineering,
The Faculty of Life Sciences and Technology,
Wroclaw University of Environmental and Life Sciences,
E-mail: deta.luczycka@up.wroc.pl

Radosław Wróbel, PhD
Institute of IT systems,
Wroclaw University of Applied Informatics,
Wejcherowska 28, 54-239 Wrocław, Poland
E-mail: rwrobel@horyzont.eu

НЯКОИ ПРОБЛЕМИ ПРИ УПРАВЛЕНИЕТО НА ЕЛЕКТРОННИ УСТРОЙСТВА С НИСКА КОНСУМИРАНА МОЩНОСТ

Сава И. Иванов, Жейно Ив. Жейнов

Резюме: В статията се разглеждат подходите за намаляване на консумираната мощност при малогабаритни автоматични електронни устройства с автономно захранване и микропроцесорно управление. Посочени са принципите при проектиране и изграждане на подобни системи, както и някои техни особености. Направен е обзор на съвременни едночипови микрокомпютри, които могат да работят в режим на намалена консумация. Показани са възможностите за намаляване на консумацията на апаратурата при използване на микроконтролери. Даден е пример за реална автономна микропроцесорна система за сбор на данни.
Ключови думи: Режим на ниска консумирана мощност, ниска консумирана мощност, микроконтролер.

Some problems of the control of hardware low-power devices

Sava I. Ivanov, Zhejno I. Zhejnov

Abstract: In the article the approaches to reduce power consumption of automatic compact electronic devices with independent power supply and microprocessor control is examined. The principles of the design and construction of such systems and some of their characteristics are listed. A survey of contemporary single-chip microcomputers that can operate at reduced power is made. The possibilities for reducing the power consumption of the equipment are shown. An example of a real autonomous microprocessor system for data collection is presented.
Keywords: Low power mode, low-power consumption, microcontroller.

1. Увод

Създаването на съвременна електронна апаратура с автономно захранване не е лека задача. Когато устройството се проектира за дългосрочна надеждна работа в полеви условия, е необходимо при разработката да се вземат мерки за намаляване на консумираната мощност и да се предвиди отделен блок за автономно захранване с използване на алтернативни токоизточници.

Първото изискване се удовлетворява чрез грижливо проектиране на електрическите вериги и използване на подходящи нови схемотехнически решения с модерни интегрални схеми и електронни компоненти с цел икономична работа, при използване на съвременни електронни елементи и управляващ микроконтролерен блок с възможност за работа в режими на ниска консумирана мощност, както и с грижливо проектиране на програмното осигуряване.

Вторият проблем се решава с използването на токоизточници, които преобразуват природна енергия в електрическа. Съвременните алтернативни захранвания зареждат специален тип акумулаторна батерия от енергията на вятъра и слънчевата светлина, когато ги има в достатъчно количество. В останалото време захранването черпи електрическата енергия на заредения акумулатор, за да се осигурят необходимите захранващи напрежения за нормалната работа на устройството. Управлението на заряда на акумулатора от алтернативните токоизточници и изработването на съответните напрежения за нормалната

работа на апаратурата се осъществява чрез внимателно проектиран контролер на заряда, като се реализира преобразуване на енергията с максимално възможен за условията КПД.

Тъй като задачите, посочени по-горе, изискват поотделно детайлно разглеждане, по-нататък ще се спрем на проблемите за намаляване на консумацията чрез подобряване на дизайна и управлението на тези устройства. Ще покажем и някои технически решения.

2. Изложение

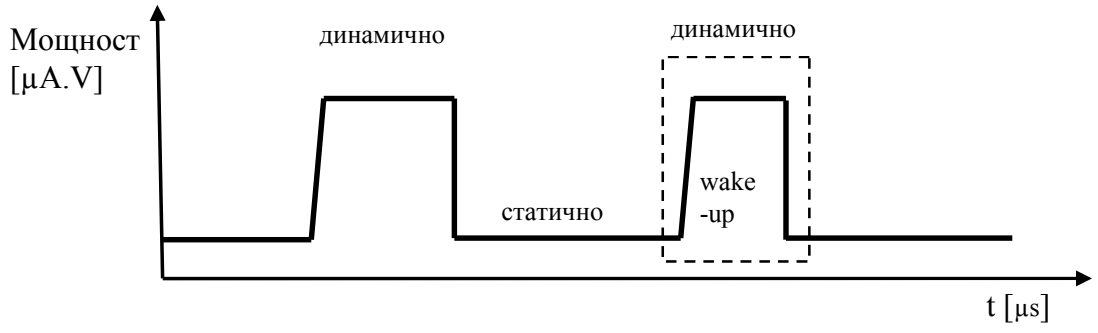
Към описания клас устройства се предявяват следните изисквания, които могат да се удовлетворят от някои специализирани микроконтролери:

- Дълъг срок на експлоатация – над 10 г;
- Висока надеждност и устойчивост на повреди: устройството трябва да следи разряда на батериите, да съобщава на ползвателя за състоянието им, а също така да обезпечава коректно изключване. Трябва да се осигури зададеният ред на изпълнение на операциите.
- Необходимо е периодично да се изпълняват определени задачи (например измерване на нива с помощта на датчик през няколко секунди).
- Наличие на режим на ниска консумация – ако при нормална работа и тактова честота 4 MHz микроконтролерът консумира 1.5 mA, то при режим на ниска консумация (напр. режим Deep Sleep на Microchip) консумацията е само 20 nA.
- Малко време за преминаване от режим на ниска консумация в реален режим (време за събуждане – Wake-Up time) - около 50 μs.

Микроконтролерът в устройството работи в два режима: динамичен (активен) и статичен. При динамичния режим системният генератор работи. Консумираната мощност е сума от паразитната консумация, консумацията на генератора, периферията, ядрото и портовете. В динамичен режим приложението е активно и изпълнява задачи. Консумацията се определя от токовете на превключване на CMOS структурите и е функция на честотата и напрежението. Периферията и портовете консумират допълнително мощност.

При статичен режим системният генератор е изключен. Консумираната мощност е сума от паразитната консумация /утечка/ в CMOS-структурите, консумацията на часовника за реално време, супервайзорите, стражевия таймер, утечките на портовете и др. подобни. Утечката расте при по-малки по размери транзистори, по-високо работно напрежение и по-висока температура.

На фиг. 1 е показана примерна зависимост на консумираната мощност във времето от режима на работа. Средната консумирана мощност е сумарната консумирана мощност за цикъл на устройството.



Фиг. 1. Консумирана мощност във времето

Проектирането на устройство с ниска консумация изисква низ от компромиси на ниво дизайн и приложение [1].

Ако се разгледа по-подробно баланса на консумираната мощност, се вижда, че той зависи от режима на работа на микроконтролера, както следва:

RUN- активна консумация:
- Такт към ядрото и периферията.
- Типичен консумиран ток 50-360 μA (3V, 25^0 C).
- LP INTRC (31kHz) – до 8 μA (1.8V, 25^0 C, PIC24F04KA201).

DOZE (за някои контролери) - активна консумация:
- Ядрото се тактува с понижена честота, периферията с пълна.
- Обикновено 35%-75% от тока в режим RUN.

IDLE (за някои контролери) - активна консумация:
- Ядрото не се тактува. Периферията се тактува и е включена.
- Обикновено 25% от тока в режим RUN.

SLEEP- статична консумация:
- Обикновено 100nA (3V, 25^0 C).
- При 85^0 C до 1.35 μA (1.8V, 85^0 C, PIC24F04KA201).

DEEP SLEEP (за някои контролери) - статична консумация:
- SRAM, VREG, VBOR, RTCC са изключени.
- Обикновено 35 nA (3V, 25^0 C).

Фирмата Microchip предлага 16 нови микроконтролери (2 семейства 8-битови и 1 семейство 16–битови), създадени по технологията nanoWatt – стандарт за всички нови микроконтролери на Microchip от 2003 г., които са идеални за устройства с батерийно захранване и имат съвместими режими на ниска консумация, периферия и развойни средства. Тези микроконтролери имат голяма периферия, микропотребление и поддържат USB интерфейс и капацитивни сензори. Усъвършенстваната технология nanoWatt XLP (eXtreme Low Power) Technology представлява набор от методи за проектиране и разработка на микроконтролери, обезпечаваща консумиран ток по-малко от 100 nA в чакащ (Sleep) режим, ток на консумация на RTCC 800nA и на WDT 800 nA.

Режимите на работа на микроконтролерите, изработени по технологията nanoWatt XLP, са систематизирани в Таблица 1 [3].

Таблица 1. Режими на работа при техология nanoWatt XLP

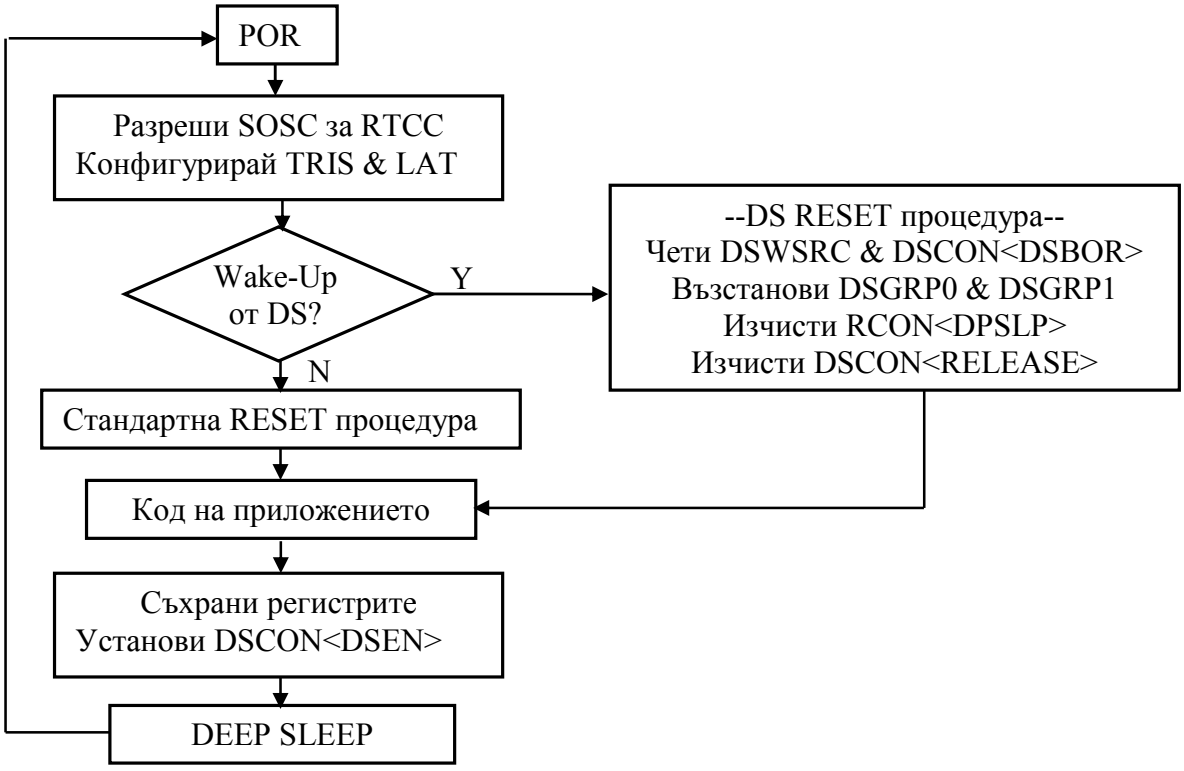
	PIC16	PIC18	PIC24
RUN Всички системи работят	+	+	+
DOZE CPU е по-бавен от периферията			+
IDLE CPU е изключен, периферията работи		+	+
SLEEP Системният такт е спрян	+	+	+
DEEP SLEEP RAM е изключена, Vreg не работи		+	+

При анализ и оптимизация на консумираната мощност времето на работа на устройството се разделя на няколко фази. Определя се тока във всяка фаза, колко време се губи на всеки работен стадий и се пресмята консумацията по времето на всяка една фаза. После се определя средната мощност на цялото устройство. Проверява се дали тя може да се намали като се намали времето за активен режим. Опитва се да се промени напрежението, тактовата честота и режима на работа за по-малка консумация на всеки един стадий на работа на устройството.

Определя се най-лошият вариант на работа и се преработва. Проверява се дали е възможно използване на други комбинация от режими на работа. После се построява графика на консумацията. Сглобява се окончателно, измерва се и се потвърждава профила на консумацията.

Цитираният по-горе режим на работа Deep Sleep, който е най-добрият енергоспестяващ режим, осигурява с 90% по-ниска консумация. При него съдържанието само на 2 регистъра за данни се запазва, а другата памет не се съхранява. Данните обаче могат да се запишат преди това във FLASH или EEPROM. Вътрешният регулатор на ядрото на микроконтролера е изключен. Излизането от режима Deep Sleep предизвиква начално установяване на микроконтролера. Повечето SFR се установяват в състояние по подразбиране. RTTC продължава да отчита времето и може да генерира секундни импулси, при консумация около 500 nA. Портовете за вход-изход съхраняват своето състояние. Запазват се състоянията на SFR регистрите DSGPR0, DSGPRB1, DSBOR [2].

Излизането от режима Deep Sleep става чрез прекъсване от вход за прекъсване, от Deep Sleep стражеви таймер (DSWDT), при аларма от RTCC, при събуждане от Ultra Low-Power Wake-Up, от сигнала MCLR или включването на захранващото напрежение (POR).



Фиг. 2. Блок-схема на цикъла Deep Sleep

Енергията, консумирана по време на цикъла Deep Sleep (DS), за инициализация и POR се дава с израза:

$$P_{DS} = (t_{DS}.I_{DS} + t_{init}.I_{init} + t_{POR}.I_{POR}).V_{supply} \quad , \qquad (1)$$

където t_{DS} е продължителността на цикъла DS, t_{init} е продължителността на инициализацията, t_{POR} е продължителността на Power On Reset процедурата. C I във формулата са означени консумираните токове по време на събитията, указани със съответните индекси.

За намаляване на консумацията от системата при проектирането се прилагат следните правила [5]:

- Всички елементи на схемата трябва да са винаги под напрежение, а управлението на захранването им трябва да става чрез индивидуални енергоспестяващи режими;
- Намаляването на захранващото напрежение намалява консумацията. В Sleep режим трябва да се направи опит за намаляването му;
- Ползва се микроконтролер с вътрешен стабилизатор на напрежение и източник с няколко изходни напрежения, ако се изискват няколко захранващи напрежения. Правилен избор на регулатор на напрежение може да намали статичната консумация;
- Четенето на RAM изисква повече мощност от четенето на FLASH. Затова е добре да се минимизира програмно достъпа да RAM;
- Резисторите към $+V_{dd}$ трябва да са с високо съпротивление;
- Ползват се кондензатори с малка утечка, къси свързващи проводници, блокировъчни кондензатори, не се оставят висящи входове;
- Кварцовият генератор стартира и генерира стабилна честота след 1024 цикъла, което може да е доста дълго време. Вътрешния RC генератор стартира за 1-5 μs. Вграденият RC генератор не е точен (до 0.25% неточност на честотата). Ползва се при нужда кварц;
- Изключва се ненужната периферия. Оптимизира се кода чрез експериментиране със скоростта, обема на кода, използваната RAM. Оценява се времето, необходимо за изпълнение на програмата;
- Ползва се SPI вместо I^2C;
- Избира се периферия, която може да продължи да работи и да събуди микроконтролера в режим Deep Sleep;
- Ползва се Deep Sleep, ако приложението не е активно дълго време - повече от 1 s, при екстремни температури, когато са нужни точни временни интервали с минимална консумация. В режим DS трябва да се ползва активна периферия.

Описаните принципи за намаляване на консумацията са приложени при проектирането на система за сбор на данни с автономно захранване и ниска консумация. Целта на разработката е натрупване на данни от датчици, монтирани в измервателен модул, като данните се обменят по RF канал. Предвидено е системата да се разполага далеч от захранваща електрическата мрежа. Консумацията и е много малка и само в момента на предаването. Ето защо захранването става от литиеви батерии с много малък саморазряд и дълъг живот. Избран е управляващ микроконтролер от вида MSP430 с интегрирано RF ядро [6]. Данните се натрупват във външна EEPPROM памет тип 24FC64 с обем 64 Kbit, също реализирана по технология с ниска консумация. Максималният консумиран ток на паметта не надхвърля 3 mA при захранващо напрежение 1.7-5.5V. В режим standby паметта консумира 1μA ток, което позволява директно свързване на захранващия извод на чипа памет с изходен порт на микроконтролера. Обменът се извършва серийно по интерфейс I^2C.

Микроконтролерът изисква кварцов кристал с честота 32768 Hz, от която се получава тактовата честота за CPU 1, 4, 8, 12, 16, 20 MHz. В активен режим на CPU микроконтролерът консумира около 160 μA/MHz на тактовата честота. Извикват се софтуерно 5 режима с ниска консумация – LPM0-LPM4, в които режими CPU се изключва. Събуждането на микроконтролера става чрез прекъсване. Програмата, обслужваща прекъсването, възстановява режима с ниска консумация, когато завърши работата си и се изпълни инструкция за връщане на управлението. Минималната консумация на системата в LPM4 при запазване на съдържанието на RAM е около 1μA.

Радиомодулът консумира 15 mA ток при предаване и приемане, като мощността на предавателя може да се регулира софтуерно, за да се постигне сигурна връзка при по-ниска излъчена мощност и съответно консумация. Той работи с отделен кварцов кристал и поддържа синхронен и асинхронен приемопредавателен стандартен радиокомуникационен протокол за съвместимост с Wireless M-bus стандарт EN13757-4:2005.

3. Заключение

Проектирането на електронни устройства с ниска консумирана мощност и автономно захранване изисква енергиен анализ и после грижливо проектиране на устройството на различни нива – архитектурно, схемотехническо, алгоритмично. Изложените по-горе принципи за намаляване на консумираната мощност касаят оптимизация на електрическите схеми и софтуера на устройства, при които режимът на работа с ниска консумация е заложен още при проектирането на управляващия микроконтролер и е реализиран чрез съответна технология, архитектурни решения и елементна база. За да се получат добри резултати е нужно както дълбоко познаване на принципите на действие и режимите на работа с ниска консумация на използвания микроконтролер, така и внимателно проектиране, тестване и настройка на управляващия софтуер с цел най-пълно използване на заложените възможности на хардуера.

Литература

[1]. Георгиев Ст., Тянев Д., Йосифов В.. Преглед на подходите и методите за проектиране на компютърни архитектури с ниска енергийна консумация. Сп. “Компютърни науки и технологии”. ТУ-Варна, бр. №2/2007, стр. 11-23.
[2]. Microchip. PIC24F Family Reference Manual S. 39. Power-Saving Features with Deep Sleep. USA, Microchip Technology Inc., 2009, pp. 5-7.
[3]. Brant, I. AN1267. nanoWatt and nanoWatt XLP™ Technologies: An Introduction to Microchip’s Low-Power Devices. USA, Microchip Technology Inc., 2009, pp. 1-6.
[4]. Dillon, J. AN1288 Design Practices for Low-Power External Oscillators. Microchip’s Low-Power Devices. USA, Microchip Technology Inc., 2009, pp. 1-3.
[5]. Microchip. Compiled Tips ‘N Tricks Guide, Ch. 2. PIC® Microcontroller Low Power Tips ‘n Tricks. USA, Microchip Technology Inc., 2009, pp. 2-12.
[6]. Texas Instruments Inc. CC430F513x MSP430 SoC with RF Core. USA, Texas Instruments Inc., 2009-2010.

За контакти:
доц. д-р Сава И. Иванов
катедра „КНТ”, ТУ-Варна
E-mail: iivanov@tu-varna.bg

гл.ас. Жейно Ив. Жейнов
катедра „КНТ”, ТУ-Варна
E-mail: zh_viv@abv.bg

ВЪЗМОЖНОСТИ ЗА ПРЕДАВАНЕ НА ДАННИ ЧРЕЗ ПОЛИМЕРНИ ОПТИЧНИ ВЛАКНА

Жейно Ив. Жейнов

Резюме: В статията се разглеждат предавателните характеристиките на един вид оптични влакна. Прави се сравнение на възможностите за предаване на данни чрез класическите стъклени оптични влакна и полимерните оптични влакна. Направен е обзор на нови конструкции на полимерни оптични влакна. Описани са профилът на показателя на пречупване, модовата структура, свързана с него, спектралните характеристики, дисперсията и затихването на влакното. Показани са предимствата и недостатъците на полимерните оптични влакна. Дадени са областите на приложение и тенденциите при разработката на този вид оптични влакна. Посочени са някои съвременни достижения в тази област.
Ключови думи: Полимерни оптични влакна, спектрални характеристики, дисперсия, затихване.

Possibilities for data transmission of the polymer optical fibers

Zhejno I. Zhejnov

Abstract: In the article the transmission characteristics of one type of fiber are examined. A comparison of the opportunities for transmission of data by traditional glass fiber with polymer optical fibers is made. A review of the new structures of polymer optical fibers is included. The profile of refractive index and the spreaded modes, associated with it, spectral characteristics, dispersion and attenuation of the polymer fibers are described. The advantages and disadvantages of the polymer optical fibers are shown. The application areas and trends in the development of this type of fiber are given. Some recent developments in this field are presented.
Keywords: Polymer optical fibers, spectral characteristics, dispersion, attenuation.

1. Увод

По-голямата част от оптичните мрежи за достъп в Европа не са изградени от традиционните комуникационни оператори, а от общините, енергийни компании и алтернативни оператори. Широколентовият достъп все още широко експлоатира медни цифрови абонатни линии (A)DSL и кабелни модеми. Средното покритие на Европа с FTTH/B в началото на 2011 г. е около 10% в сравнение с 30% другаде. В България има изоставане на въвеждането на този тип мрежи в селските по сравнение с градските райони [10].

В същото време интернет сърфистите с всеки изминал ден все по-интензивно свалят музика и филми. Ползването на P2P услугите бързо създава в мрежата „тесни места”. Ежедневно симетричната им природа употребява около 80% от честотната лента в потока от абоната. За да се справят с растящия трафик, телекомуникационните оператори се нуждаят от икономична и перспективна технология, която да бъде съединително звено между градските мрежи и крайните потребители. Лентата на пропускане трябва да се разпредели между отделните домове и апартаменти, които обикновено са на разстояние не повече от 300 m от най-близкия мрежов възел (разположен напр. на края на тротоара или в мазето на многоетажно здание). Тази част от мрежата изисква най-големи капитални разходи и притеснява най-много операторите.

2. Изложение

В началото на 2006 г. 9 европейски компании и изследователски институти започнаха да създават технология, способна да замени кварцовите влакна при строителството на FTTH мрежи, като значително ще намали нейната цена. Двугодишният проект POF-ALL (Paving the Optical Future with Affordable, Lightning-fast Link) с бюджет 1,6 млн. € имаше за цел усъвършенстването на стандартното стъпално полимерно оптично влакно (POF) на основата на poly-methil-metha-acrilate – PMMA и използването му за работа в оптична система за връзка с FTTH мрежите на скорости от 100 Mbit/s до 1 Gbit/s на разстояния 100-300 m. За съобщителна среда вместо стандартно стъклено влакно се ползва POF с голям диаметър. POF лесно се свързва и терминира с обикновени инструменти, предава видимата светлина и има висока еластичност. Полимерните влакна имат външен диаметър 0,5-1 mm, което значително облекчава монтажа и работата с него (Табл. 1).

Таблица 1. Диаметри на сърцевината/обвивката за стъклени и полимерни влакна

Влакно	SMF	MMF	Кварцово	POF
Сърцевина[μm]	9	62,5	200	980
Обвивка [μm]	125	125	300	1000

Технологията POF получи известност поради широкото разпространение в автомобилните информационни мрежи, основани на стандарта MOST и ByteFlight. Днес над 30 млн. MOST трансивъра са разположени по 60 различни модели машини на производители като BMW, Daimler-Chrysler, Porsche, Audi и Saab. Високата надеждност на POF системите в тази област намали цените на съответните трансивъри до 2-4 € [2].

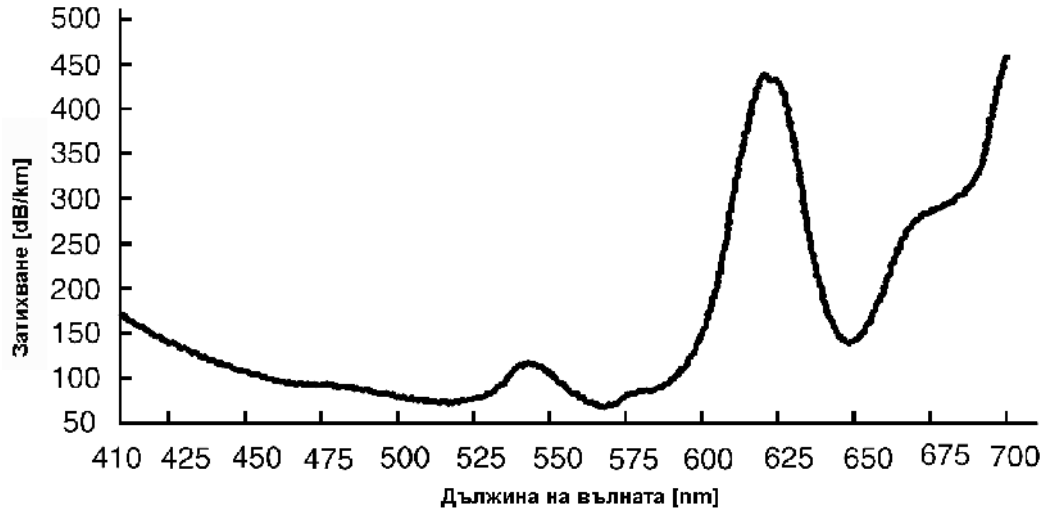
Най-ниската реализирана пропускна способност 100 Mbit/s на модем за POF, е 20-100 пъти по-висока от скоростта на DSL-връзка и позволява зареждане на филм с DVD качество за по-малко от 3 min и реализация на услуга VoD (Video on Demand) в реално време. POF поддържат еднакви скорости на трансфер във възходящо и низходящо направление, което прави възможни видеоконференции и предаване P2P на любителски филми. Максималното разстояние 100-300 m удовлетворява изискването към крайните мрежи FTTH в европейските градски райони за разстояние между оптичното влакно, прекарано до фундамента на зданието и цифровия извод в отделните апартаменти. 85 % от крайните мрежи в Италия например отговарят на стандарта FastWeb и имат дължина под 300 m. Подобна е ситуацията в повечето европейски страни.

Технологията POF-ALL може да се ползва за изграждане на широколентова домашна мрежа с всички предимства на по-скъпите и сложни оптични съединения, изградени с кварцови влакна. Всеки сам може да монтира POF за по-малко от 30 min., ако има нужните инструменти. За разлика от стъкловлакнестите съединения, които изискват уморително лепене и изсушаване, POF се реже с ножица, обвивката се зачиства с обикновени зачистващи инструменти, а муфата на металния съединител се поставя върху обвивката на влакното с притискащи инструменти. За POF конекторите не е страшен останалия прах върху челните повърхности, (което е един от основните проблеми на съединяването на стъкловлакната). Не се изисква щателно почистване и полиране. Някои устройства работят и без съединител. POF може да се отреже с нож и да се постави на нужното място, което е много по-евтино от монтажа на стъкловлакното от специално обучен персонал. Възможно е да се инсталира самостоятелно кабелна мрежа в собственото здание.

POF кабелите имат висока здравина и издържат на удари, без да губят еластичността си, което е необходимо за монтаж по стените. Друго предимство е използването на видима светлина вместо инфрачервена. Това изключва риска за изгаряне на роговицата и слепота, който съществува при работа с кварцово влакно. В домовете не бива да се ползват

стъкловлакна, защото деца биха могли да се наранят, ако погледнат в куплунга на трансивъра. Видимата светлина позволява и визуално тестване на влакното, което спестява значителни суми при настройката на мрежата.

Усъвършенстването на технологията и дизайна на полимерните влакна позволиха известно разширяване на честотната лента, но затихването на влакното във видимия диапазон, обусловено от собственото поглъщане в материала и взаимодействието на светлината с хармоничните съставящи на честотата на трептенето на въглеводородните групи, е много високо. Най-малко затихване от старите влакна дава полиметилметакрилатното (PMMA) влакно. За трансфер съответно при 100 Мбит/с на 300 m то е около 140 dB/km при дължина на вълната 650 nm и при 1 Gbit/s на 100 m е около 80 dB/km на 520 nm /фиг. 1/.



Фиг. 1. Затихване на PMMA POF като функция от дължината на вълната

За реализация на предаването е необходима минимална мощност на предавателите 24 dBm за първия вариант и 14 dBm за втория вариант. Лентата на пропускане за стандартно POF със стъпален профил на показателя на пречупване обикновено е ограничена на 30 MHz при 100 m влакно. С обикновените схеми на кодиране не могат да се постигнат скорости 100 Mbit/s и повече. Ползват се схеми на кодиране с много нива за осигуряване на необходимите съчетания дължина на влакното и скорост на трансфера [8]. На големите скорости е по-добре да се ползва градиентно POF, но са възможни и други решения.

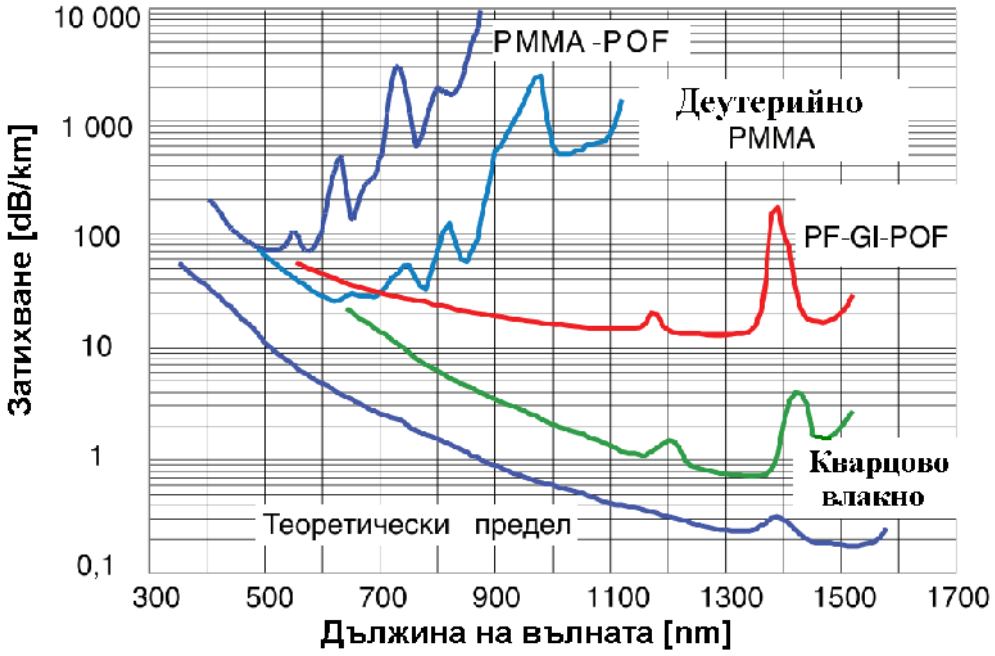
Намаляване на загубите се оказва възможно чрез замяна на водорода в материала на влакното с изотопа му деутерий. Това обаче е много скъпо и не се прилага. Чрез замяна на водорода с флуор през 2005 г. са получени градиентни перфлуорполимерни влакна (PF-GI-POF), в които загубите са намалени до 50 dB/km в диапазона 650-1300 nm. Теоретичният минимум на загубите в такова влакно е около 10 dB/km. Следващата перспективна разработка е създаването на микроструктурирано кухо полимерно оптично влакно [9].

Коефициентът на затихване се определя от разсейването и поглъщането на светлината във влакната. Разсейването може да става от нееднородностите на материала, от обемни дефекти, както и на границата с обвивката или на изходните повърхности на влакното. Загубите, обусловени от тези механизми, зависят от технологията на производството и могат силно да се намалят при нейната оптимизация. Допълнителни загуби внасят примесите и фундаменталното поглъщане. От въглеводородните полимерни влакна най-малки загуби има полиметилметакрилатното (PMMA). То има два работни прозореца около 570 нм и 650 nm. Възбуждането на влакното на тези вълни става с евтини светодиоди (LED) и червени

полупроводникови лазери. Високото затихване (в момента минимумът е около 70 dB/km) ограничава разстоянието за предаване на информация в такова влакно до 100 m.

Аморфните перфлуорполимери имат спектър на пропускане 600-1300 nm, а загубите са по-малки от 50 dB/km. Снижаването на загубите чувствително увеличава далечината на връзката. Различният диапазон на работа на влакното пък позволява използване на по-евтино оборудване, което вече е разработено за кварцово оптично влакно (800-1300 nm).

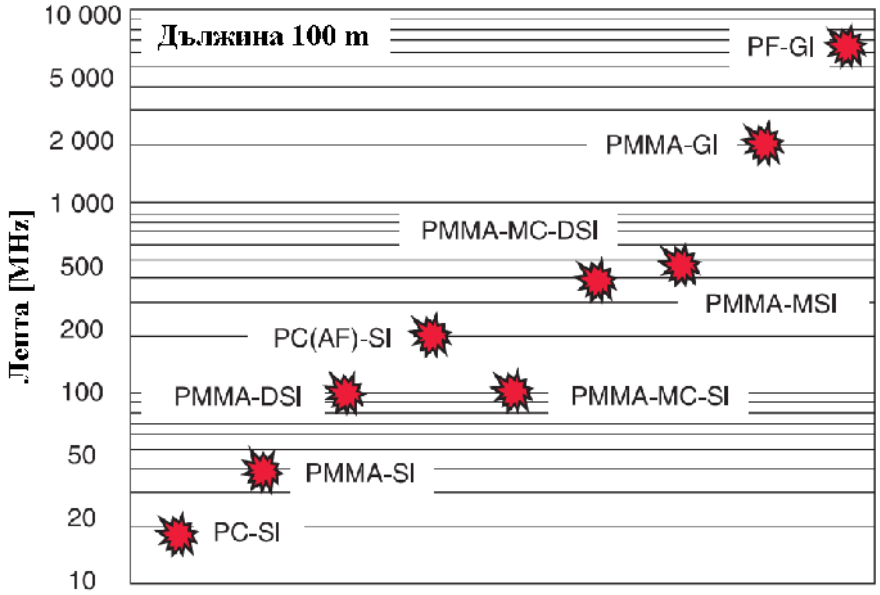
PF-POF при дължина на вълната 1300 nm внасят загуби под 20 dB/km, при 800 nm загубите са под 30 dB/km, а при 650 nm те са до 60 dB/km. Минималните загуби, получени при този тип влакна, са при 1300 nm – 15 dB/km при теоретичен минимум под 1 dB/km. Тези резултати обаче са за влакна с диаметър 100-150 μm [3]. Сравнение на спектрите на поглъщане на изброените по-горе полимерни оптични влакна и стъклените, както и на SMF такова е показано на фиг. 2.



Фиг. 2. Спектри на поглъщане при различни оптични влакна

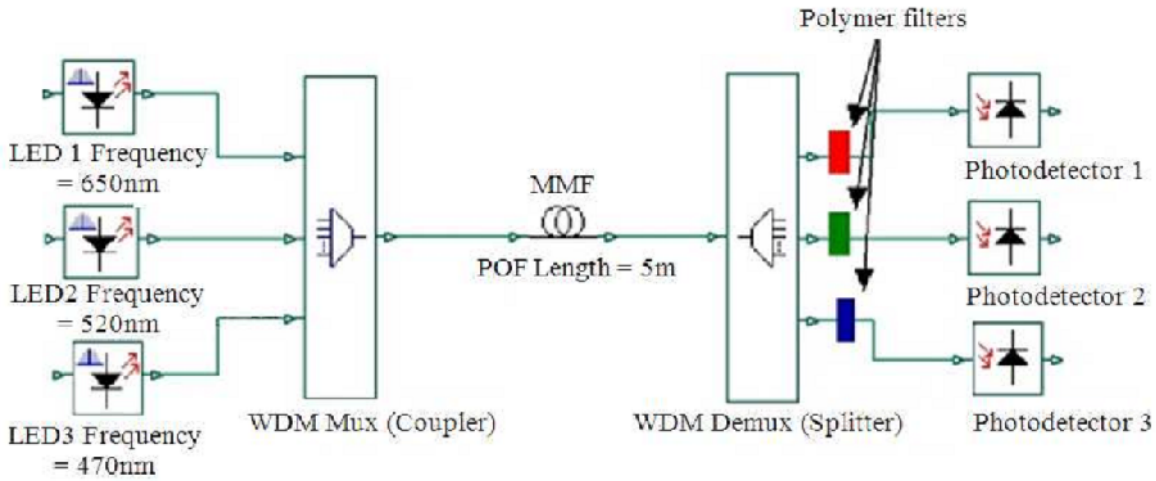
Повечето POF имат обикновено стъпален профил на показателя на пречупване (SI). В работния диапазон този вид влакна работят като многомодови и имат най-малка лента на пропускане. Тяхно предимство е лесната изработка. Материалът на сърцевината и разликите между коефициентите на пречупване сърцевина/обвивка влияят върху лентата на пропускане на влакното. Влакно с двустъпален профил на показателя на пречупване (позволяващо огъване при монтажа при малки загуби) е с по-тясна лента на пропускане. Втората оптическа обвивка обезпечава по-ефективно насочване на светлинния сноп в участъка на огъване, а вътрешната обвивка, която е с по-малък показател на пречупване, намалява модовата дисперсия. По-нататъшно разширение на лентата на пропускане става възможно чрез конструиране на POF с няколко сърцевини (Multi-Core), което може да бъде с едно- или двустъпален профил на показателя на пречупване. Най-широка лента на пропускане осигурява градиентното (GI) влакно с параболичен профил на показателя на пречупване. Максимално широка лента на пропускане е достигната при PF-GI-POF влакната. При дължина от 100 m такова влакно осигурява скорост на трансфер 10 Gbit/s.

Полимерното оптично многостъпално влакно обезпечава почти същата лента на пропускане, но е по-просто за изработка. Сравнение на лентата на пропускане за влакна с еднаква дължина и различен дизайн, които са изработени от различен материал, е показано на фиг. 3.



Фиг. 3. Лента на пропускане на влакна от различен материал и с различен профил на пречупване.

Модел на локална оптична мрежа на основата на POF е предложен в [6]. Той реализира евтина 3-канална WDM система за предаване на данни, аудио и видео сигнал по полимерно PMMA влакно с дължина 5 m /фиг. 4/.



Фиг. 4. Блок-схема на проста POF-WDM оптична мрежа

Светлинни източници са LED с различен цвят. Използват се мултиплексор и сплитер за предаване на Ethernet (данни), видео и аудио на 3 различни дължини на вълната на светлината: съответно 470 nm (синя), 510 nm (зелена), 665 nm (червена). Приемниците са силициеви фотодиоди.

3. Заключение

Понастоящем POF се прилагат за управление на производствени процеси и в автомобилостроенето. Търсенето им нараства и се диктува от нуждите на каналите за предаване на данни, защитени от смущения. POF са много подходящи за галванично разделяне на управляващи и нискосигнални устройства от мощна апаратура с високо напрежение – електрожени, рентгенови апарати, оборудване за йонно легиране. Тези влакна представляват особен интерес за автомобилните иновации. Работи се за значително

намаляване на цените на трансивърите за шината MOST, както и увеличаването скоростта и на трансфер от 25 Mbit/s на 150 Mbit/s. Разработени са два нови вида POF автомобилни мрежи (ByteFlight, Flextray), които се нуждаят от съответна апаратура. В САЩ и Япония автомобилните фирми планират въвеждане на системи, работещи на скорости 400 Mbit/s и стандарт за тях – ID-1394. Разработените евтини трансивъри за автомобилната MOST мрежа и други оптични елементи масово ще се използват за изграждане на домашни и локални мрежи. Малките размери, тегло, устойчивостта на удари и вибрации, високата скорост на предаване на данни поставят въпроса за навлизането на POF във военни проекти и авиацията. Разработват се влакна, устойчиви на високи температури и негорими такива. POF са перспективни за големи центрове за обработка и предаване на данни и за съединяване на група сървъри и суперкомпютри. Те консумират по-малко енергия от медните кабели при работа. Големи компютърни фирми като например Intel, изследват възможностите за използване на оптична шина на основата на POF за обмен на данни с дънната платка на компютъра. Оптичните шини поддържат по-високи скорости на обмен от аналогичните електрически и лесно се интегрират в печатната платка и активната електроника. Конструкторите активно търсят възможности за ползване на оптични съединения в системните платки на компютри, маршрутизатори, превключватели, памети и др. типове електронни модули, както и за свързване на модули, стойки и възли в центровете за обработка на данни.

Литература

[1]. Kuzyk, Mark G. Polymer Fiber Optics. Materials, Physics, and Applications. Taylor&Francis Group, Boca Ration, FL, 2007.

[2]. Наний, О.Е., Е.Г. Павлова, И.А. Таначев. Полимерное оптическое волокно: достижения и перспективы практического применения. //Lightwave Russian Edition, 2007 № 4, стр. 31-34.

[3]. Ночивелли, А.. Полимерные волокна – универсальный оптический доступ. //Lightwave Russian Edition, 2006 № 3, стр. 4-6.

[4]. Les, C.B.. Australian Agenda – No Great Barriers. //Photonics Spectra 2009 №9, pp.66.

[5]. Haynes, R., J. Bland-Hawthorna, M.C.J. Large, K.F. Kleinc. New age fibers: the children of the photonic revolution. Proceedings of the SPIE, Volume 5494, 2004, pp. 586-597.

[6]. Fischer, U. H. P., M. Haupt, M. Joncic. Optoelectronics - Devices and Applications, Ch. 22. InTech, Rijeka, Croatia, 2011, pp. 445-468.

[7]. Ab-Rahman, M. S., Farshad N. K.. Cost-Effective Wire-Harness Model by Using Polymer Optical Fiber. Journal of Computer Science 9 (7), 2013, pp. 935-942.

[8]. Lee, S. C. J.. Discrete Multitone Modulation for Short-Range Optical Communications. Eindhoven: Technische Universiteit Eindhoven, 2009, pp. 95.

[9]. Argyros, A., I.M. Bassett , M.A. van Eijkelenborg , M.C.J. Large , R.C. McPhedran. Air-core photonic band-gap fibres without polarisation degeneracy. 2003 Digest of the LEOS Summer Topical Meetings, Vancouver, BC, Canada, 2003.

[10]. IDATE Consulting & Research. Broadband Coverage in Europe. Final Report 2011 Survey. DG INFSO 2011.

За контакти:
гл. ас. Жейно Ив. Жейнов
катедра „КНТ”
ТУ-Варна
E-mail: zh_viv@abv.bg

ИЗСЛЕДВАНЕ НА ПРОИЗВОДИТЕЛНОСТТА НА КЛЪСТЕРНА СИСТЕМА ПРИ ИЗПЪЛНЕНИЕ НА ВИРТУАЛНА МАШИНА

Делян Х. Сърмов

Резюме: Съществуват множество задачи, решаването на които поставя високи изисквания към производителността на системата. Традиционно разходно-ефективните решения се свързват с реализация на клъстерни системи. В тази статия е изследвано изменението на производителността на клъстерна система, при която възлите на клъстера са реализирани в Host операционна система (ОС), докато наличният софтуер на компютъра се използва посредством виртуална машина.
Ключови думи: клъстер, виртуална машина, линукс, OpenMPI, KVM, NAS parallel benchmarks

Study of performance of cluster system in implementing the virtual machine

Delyan H. Sarmov

Abstract: There are many problems, resolution of which places high demands on system performance. Traditionally, cost-effective solutions are associated with the implementation of cluster systems. This article studied climate performance cluster system where the nodes of the cluster are implemented in the Host operating system (OS), while the available software on the computer is used via the virtual machine.
Keywords: cluster, virtual mashine, linux, OpenMPI, KVM, NAS parallel benchmarks

1. Увод

Персоналните компютри работят под управлението на ОС. Всяка потребителска операционна система е подложена на рискове по отношение на сигурността и стабилността, в зависимост от използваните приложения и мрежови услуги. Може да се каже, че познанията и опита на потребителя, работещ на нея, също е важен фактор за стабилността на системата. При реализация на клъстер трябва да се отчетат тези фактори и да се минимизира риска от грешки в изчисленията, предизвикани от събития в потребителската операционна система. С нарастване на броя на възлите в клъстера пропорционално се увеличава и възможността за грешки поради неправилна работа на потребителската операционна система.

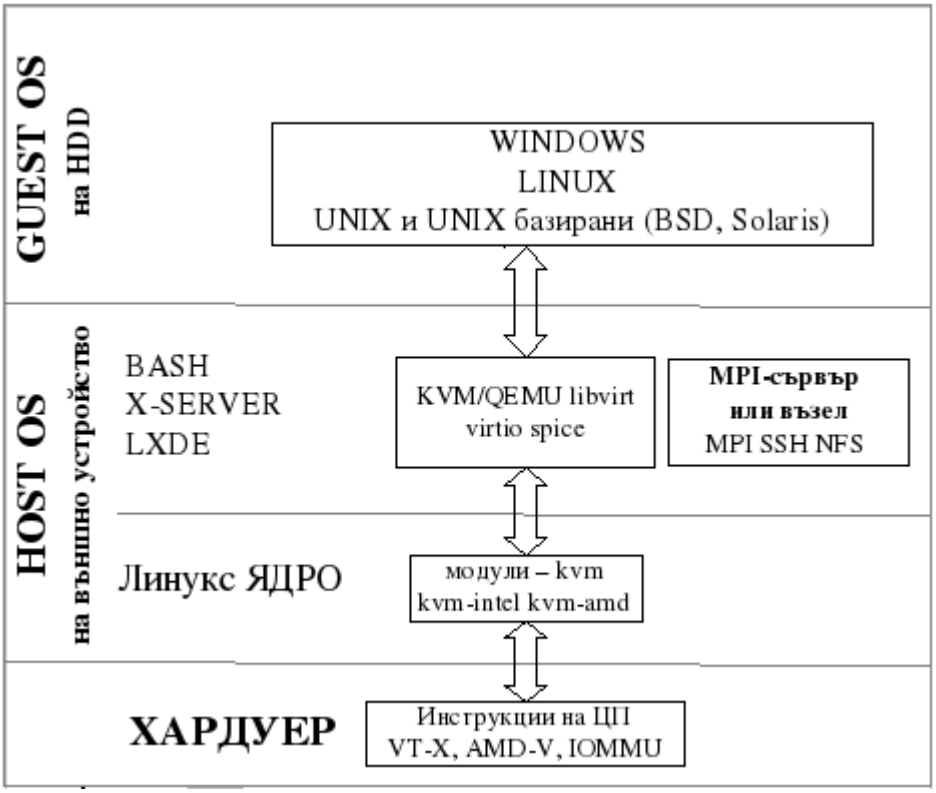
За да се минимизира възможността от грешки, предизвикани от работата на потребителската система, е използвана изолация на ниво операционна система. Благодарение на бързото развитие на виртуалните машини вече съществуват инструменти, с които това може да бъде постигнато. Необходимо е да се изследва производителността на клъстера, когато част от хардуерния ресурс е зает с изпълнението на виртуална машина.

2. Изложение

При изграждане на клъстера се използва GNU/Linux в качеството на операционна система Host. Изборът се мотивира с необходимостта ОС да притежава инструменти за качествен контрол на натоварването, настройка и мониторинг на системата. Необходимият софтуер за изграждане на клъстер с използване на MPI съществува и е оптимизиран за използване под управление на GNU/Linux [1]. Не е без значение и финансовата страна. Използването на ОС с отворен код позволява да се спестят средства за закупуване на лицензи, които са необходими при комерсиалните ОС. GNU/Linux може лесно да бъде

адаптирана за инсталация на преносимо външно устройство. Наличен е софтуер за виртуализация, с който може да се стартира операционната система, инсталирана на стандартното дисково устройство, като най-често това е твърдият диск.

На фиг. 1 е показана структурата на една компютърна система, която реализира възел от клъстер. При необходимост да се използва компютъра за нужди извън клъстера, софтуерът на компютъра може да се използва посредством виртуална машина.



Фиг. 1. Блок схема на един компютър от MPI-клъстер и виртулна машина

В следващите експерименти са използвани 16 еднакви двуядрени компютърни системи с 4 GB оперативна памет. Виртуалната машина е конфигурирана по следния начин:

Таблица 1. Параметри на виртуалната машина

Централен процесор	Използва се реалния процесор, от който за виртуалната машина е заделено първото ядро.
Оперативна памет	2 GB
BIOS	SeaBIOS версия 1.7
Видеокарта	VGA compatible controller: Red Hat, Inc.
Звуков адаптер	Intel 82801AA AC97
IDE контролер	IDE controller
Дисково устройство	Целият твърд диск е предоставен на виртуалната машина за монополно използване
Мрежова карта	Virtio network controller

От Таблица 1 се вижда, че за виртуалната машина е заделена половината от оперативната памет, която притежава реалният компютър. От централния процесор се използва първото ядро, но трябва да се отбележи, че програми като virt-viewer и spice притежават елементи, работещи на реалния компютър. По този начин неизползваното от виртуалната машина ядро се натоварва до 12%. Общото натоварване на двуядрения процесор е 56%.

В следващия експеримент ОС, работеща във виртуална машина, се натоварва до степен да използва максималния заделен за нея ресурс. За целта се стартират няколко приложения, работещи едновременно:

- поточно видео – за натоварване на мрежовата връзка и видео подсистемата;
- glxgears – натоварване на видео подсистемата;
- копиране на папка с 150 000 файла – за натоварване на дисковата подсистема;
- Super Pi – програма, която изчислява π с точност до 32 милиона знака след десетичния разделител. Програмата се използва за натоварване на ЦП във виртуалната машина до 100%.

Изброените програми са стартирани във виртуалните машини на 16-те компютъра, използвани като възли в клъстера. При тези условия са изпълнени тестовете за измерване на производителността на клъстера с работеща виртуална машина. За сравнение са дадени резултатите без включена виртуална машина.

2.1. Производителност на дисковата подсистема

Таблица 2. Резултати на bonnie++ без VM и с включена VM

	Последователен запис						Последователно четене				търсене	
	Символи		блокове		презапис		символи		блокове			
	K/sec	% CPU	K/sec	% CPU	K/sec	% CPU	K/sec	% CPU	K/sec	% CPU	K/sec	% CPU
без VM	23420	26	22541	3	12822	1	103186	100	47439	5	220.9	0
с VM	13955	15	11778	1	6951	1	64831	67	29812	12	122.4	0

Както се вижда от Таблица 2 производителността на дисковата подсистема намалява значително при всички операции. Цялостното натоварване на ЦП и входно-изходната система във виртуалната машина оказва влияние върху резултатите в Host OS.

2.2. Мрежов трансфер

Таблица 3. Производителност на мрежовата подсистема измерена с netperf

Recv Socket	Send Socket	Send Message	Elapsed	Throughput
Size bytes	Size bytes	Size bytes	Time	10^6bits/sec
87380	16384	16384	10,34	94,14
87380	16384	16384	10,25	94,04

За времето на изпълнение на netperf във виртуалната машина се наблюдава спад в скоростта на мрежовия трафик. За осъществяване на връзка между Host OS и Guest OS се използва мостов интерфейс, в който заявките на Host OS имат приоритет. Както се вижда от резултатите в Таблица 3, скоростта на мрежовата връзка в Host OS не се влияе съществено от работата на виртуалната машина.

2.3. Скорост на операции с оперативната памет

Централните процесори са много по-бързи от контролера на паметта в компютърни системи. Програмите често са ограничени в бързодействието си от пропускателната способност на паметта на системата, а не от изчислителна производителност на процесора.

В този тест скоростта на операции с оперативната памет се измерва с програмата stream. Програмата е специално проектирана да работи с масиви от данни, по-големи от наличния кеш на която и да е система, така че резултатите да са показателни за изпълнението на приложения, обработващи вектори. Резултатите, които се извеждат, са за четири типа операции [2]:

a(i) = b(i) - Копиране (Copy) от една клетка на паметта в друга. Обработват се блокове от 16 байта ОП;

a(i) = q*b(i) - Скалиране (Scale), при което копираните стойности се умножават с число. Обработват се блокове от 16 байта ОП;

a(i) = b(i) + c(i) – Сумиране (Sum), където стойността на клетката се получава като сума на стойностите на други две клетки. Обработват се блокове от 24 байта ОП;

a(i) = b(i) + q*c(i) – Функцията Triad комбинира операциите сумиране и скалиране. Обработват се блокове от 24 байта ОП.

В Таблица 4 и Таблица 5 са изнесени статистиките от теста stream, проведен без и с използване на виртуална машина.

Таблица 4. Резултати от теста stream без използване на VM

Функция	Скорост (MB/s)	Средно време (секунди)	Минимално време (секунди)	Максимално време (секунди)
Copy	2757.4830	0.0117	0.0116	0.0122
Scale	2795.7367	0.0116	0.0114	0.0121
Add	3020.7900	0.0160	0.0159	0.0161
Triad	3075.3317	0.0157	0.0156	0.0158

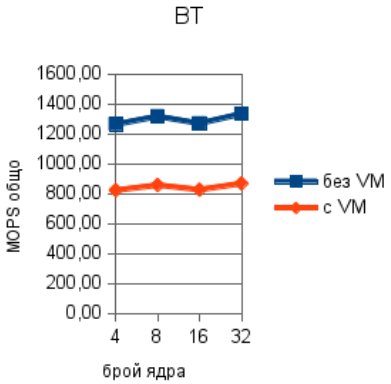
Таблица 5. Резултати от теста stream с използване на VM

Функция	Скорост (MB/s)	Средно време (секунди)	Минимално време (секунди)	Максимално време (секунди)
Copy	1999.6384	0.0169	0.0160	0.0177
Scale	1988.6759	0.0169	0.0161	0.0193
Add	2375.0586	0.0215	0.0202	0.0236
Triad	2399.6590	0.0209	0.0200	0.0215

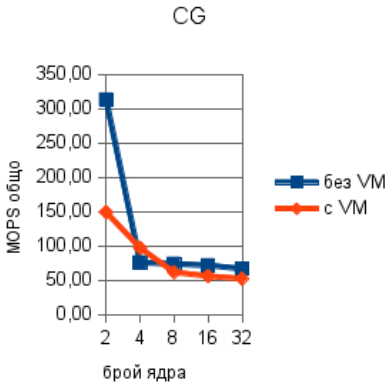
При 16 байтови блокове памет резултатите при използване на виртуална машина са значително по-ниски в сравнение с използването на 24 байтови блокове памет. Тестът е показателен за определяне на слабостите в системата. Централният процесор запазва висока производителност след стартирането на виртуалната машина, докато функциите използващи само операции с оперативната памет показват сравнително ниски резултати. С увеличаване на тежестта на входно-изходните операции в тестовете се наблюдава влошаване на производителността.

2.4. Тестване на клъстера с използване на NAS Parallel Benchmarks

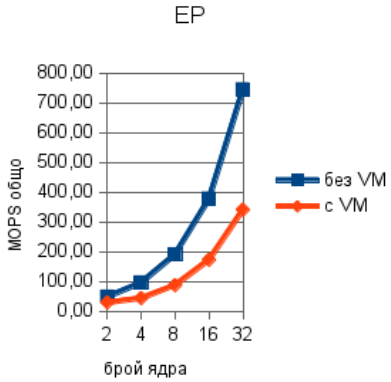
Тестът NAS Parallel Benchmarks е изпълнен с проблем от клас A последователно на 2, 4, 8, 16, 32 ядра. От фигура 2 до фигура 9 са представени графично резултатите от изпълнението на всяка една от тестовите програми включена в клас A на NAS Parallel Benchmarks. Тестовете са изпълнени със и без включена виртуална машина.



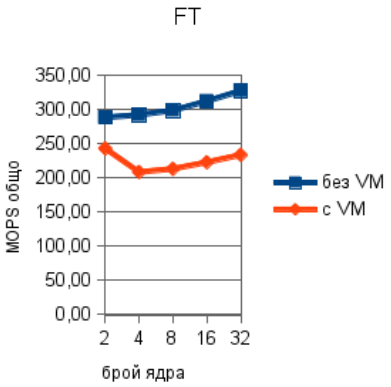
Фиг. 2. Block Tridiagonal



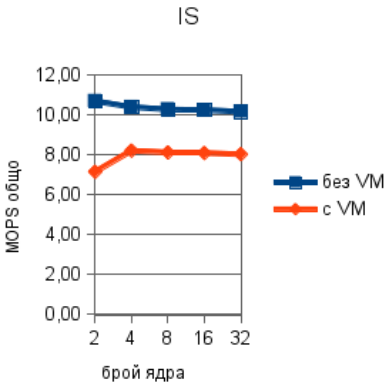
Фиг. 3. Conjugate Gradient



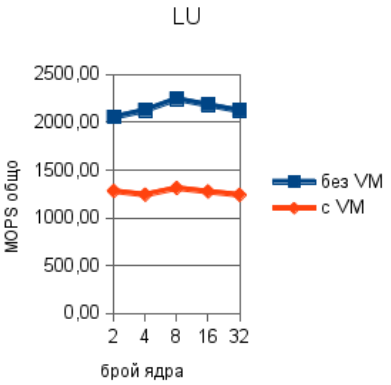
Фиг. 4. Embarrassingly Parallel



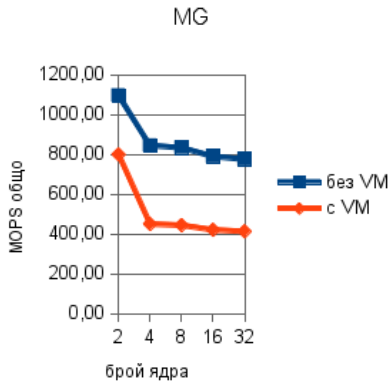
Фиг. 5. Fast Fourier Transform



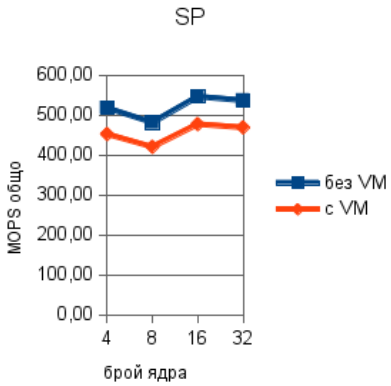
Фиг. 6. Integer Sort



Фиг. 7. Lower-Upper symmetric Gauss-Seidel



Фиг. 8. Multigrid



Фиг.9. Scalar Pentadiagonal

Натоварването на централния процесор достига 56% без стартирането на изчисления в клъстера. В резултатите, обаче при нито една от изпълнените програми не се наблюдава понижение на производителността с 56%. Само при тестовете EP, MG се наблюдава рязко понижение на производителността при включване на виртуална машина. Тези тестове са специално разработени да извлекат максимума от възможностите на паралелна ВПС [3,4], но дори и при тях производителността е над очакваната. В останалите тестове като например CG, IS и SP, производителността намалява в значително по-ниски граници.

3. Заключение

Въз основа на проведените експерименти може да се обобщи, че клъстерната система запазва работоспособност след включване на виртуална машина. Натоварването на централния процесор от виртуалната машина достига 56%, докато оставащото процесорно време за клъстера е 44%. В зависимост от тестовата програма резултатите показват производителност в интервала 47-87%, спрямо резултатите без използване на виртуална машина. Наблюдава се повишено оползотворяване на централния процесор, изразяващо се в резултати, надхвърлящи разполагаемия ресурс. Необходими са допълнителни изследвания на влиянието на използвания мениджър на процеси (Completely Fair Scheduler) [5,6], както и изследване на призводителността при употреба на алтернативни мениджъри на процеси.

Литература

[1]. William G., E. Lusk, A. Skjellum. Using MPI Portable Parallel Programming with the Message-Passing Interface, MIT press, Cambridge, Massachusetts, second edition, 1999.
[2]. Dongarra, J., P. Luszczek, "Introduction to the HPC Challenge Benchmark Suite," ICL Technical Report, 2005.
[3]. Prieto M., R. S. Montero, Ign. M. Llorente, A Parallel Multigrid Solver for Viscous Flows on Anisotropic Structured Grids, Institute for Computer Applications in Science and Engineering, NASA Langley Research Center, October 2001
[4]. NAS Parallel Benchmarks documentation, URL: <http://www.nas.nasa.gov/hecc/support/kb/22/>, 22.08.2013.
[5]. CFS Scheduler release information, URL: <http://git.kernel.org/?p=linux/kernel/git/next/ linux-next.git;a=blob;f=Documentation/scheduler/sched-design-CFS.txt>, 22.08.2013.
[6]. Kobus J., Szklarski's R., Completely Fair Scheduler and its tuning, Based on R.Szklarski's MSc. thesis, Toruń, 2009, http://www.fizyka. umk.pl/~jkob/prace-mag/cfs-tuning.pdf

За контакти:
д-р Делян Христов Сърмов
катедра „Компютърни системи и технологии"
ШУ Епископ Константин Преславски
E-mail: dsarmov@mail.bg

КВАНТОВ КОНСТРУКТОР

Николай Райчев, Елена В. Рачева

Резюме: Тази статия се фокусира върху реализацията на квантов конструктор за съставяне и изпълнение на гъвкави квантови схеми и алгоритми. Изследването показва различни операционни взаимовръзки, които могат да бъдат класически симулируеми. Приложението използва малък квантов регистър, с размер до 14 кюбита. Този експеримент показва как общите инстанции на даден проблем могат да бъдат кодирани в лесно приложими гейтове. Основната цел е създаването на изображения, които могат да помогнат за разбирането на квантовите вериги и може би ще послужат като градивен елемент за генериране на нови квантови алгоритми.
Ключови думи: симулация, квантови алгоритми, изчислителна мощност

Quantum Constructor

Nikolay Raychev, Elena V. Racheva

Abstract: This paper focuses on new realization of an simulation on implementation of quantum circuits. The research shows various concatenations and other circuits to be classically simulatable. There is a new approach into solving problems using quantum resources. It is referred to a small quantum register up to about 14 qubits. This experiment shows how the general instances of the given problem can be encoded into an easily applicable gates. Its main intention is to create images — images which may help to learn and understand quantum circuits, and which perhaps will serve as building blocks for inventing new quantum algorithms.
Keywords: simulation, quantum algorithms, computation

1. Въведение

Има няколко причини, поради които квантовите компютри трябва да бъдат проучени. Първо, квантовите изчисления скоро ще се превърнат в революционно нова технология. Квантовите компютри ще бъдат значително по-бързи от класическите компютри на нашето време. Новите технологии винаги предлагат нови перспективи, които са свързани с най-добрите интелектуални ресурси на обществото и са ключът за устойчиво развитие на науката. Квантови компютри с регистър от около 1 KQ (1000 кюбита) ще могат да дешифрират за кратко време всеки активен публичен ключ на всяка класическа криптографска система. Реализацията на квантовите компютри може да продължи едно или повече десетилетия и да направи квантовото изчисление проблем (както и намирането на решение, тъй като предлага нова криптография!) за електронната търговия на фирми и разузнавателни служби. Това е икономически и политически фактор, който е от голямо значение. Друга причина за създаването на квантови компютри са техните научни и теоретични аспекти. Квантовите компютри могат да изпълняват ролята на лаборатории за изследване на квантовите ефекти. Всяка функция може да бъде проучена обстойно, тъй като квантовите изчисления се основават главно върху: "Квантовото оплитане", при което два отделни кюбита могат да бъдат приведени в обща суперпозиция, наричана заплитане. Ако атрибут на една от частиците бъде измерен (например, неговото завъртане), след това ако друга частица има обратна стойност, без значение колко далеч са те, измерването на стойността на променливата може да бъде напълно неопределено, след което те са антикорелативни. Това явление е било противоречиво, поради откритието на Айнщайн и неговите колеги Подолски и Розен [1], които са смятали, че са намерили невъзможно

следствие на квантовата механика и по този начин да се фалшифицира контрапримера на тази теория. Това явление е било демонстрирано експериментално през 1980 година. В днешно време почти никой не се съмнява в реалността на заплитането, но философските последици все още не са ясни. Какво са квантовите изчисления? Класическата информация базира бита само на две възможни стойности, докато квантовата информация се позовава на кюбити: кюбитът апроксимира концепцията, тъй като той може не само да съдържа две стойности |0> и |1>, а също така и "суперпозициите" от тях, т.е., векторни суми като |0> + |1> или | 0> - | 1>. Това е радикално нов логичен метод, напълно непознат при класическата обработка на информация. Той дава възможност за нови видове логически алгоритми, работещи за квантовите регистри, като подредени колекции от няколко кюбити. Подобно на класическото изчисляване, съществува универсален набор от елементарни гейтове, който е достатъчен да задоволи всичките ни изчислителни нужди [3]. Стандартна техника в проектирането на квантови алгоритми е използването на *Oracle* оператор за разиграване на фазова *kickback* операция [2].

Въпреки че изчислението на цялата верига квантови алгоритми може да бъде много некласическо, то входът и изходът трябва да бъдат класически дефинирана информация, кодирана от битове [4]. Това е очевидно, защото искаме алгоритми за решаване на въпросите, но разполагаме с "класически" методи, затова се нуждаем от информация в класическа форма, тъй като само това отговаря на нашите въпроси. За да получим резултат, на изхода на квантовия регистър трябва да се направи измерване. То е единственият начин да се получи информация от класическа квантова система. Да обобщим: квантовият компютър има битове като ресурс, той работи с кюбити, и дава битове като изходен резултат.

2. Постановка на задачата

Целта на това изследване е да се направят конкретни и изрични елементи от квантовите изчисления, които типично се третират като специални случаи и изключения: обобщени оператори и кодиране на класическа информация във фазата на квантови състояния. По този начин ще се разработи общ квантов конструктор за работа с квантови вериги, който ще се характеризира с решенията на следните функции:

1. Виртуализирано обобщение на единични кюбитови и контролирани оператори, базирани на виртуализиран единичен кюбитов оператор.
2. Характеризиране на елементарни оператори като линейна комбинация от два логически базови оператора.
3. Йерархия от елементарни базови оператори за n кюбитови пресмятания. Характеризиране като булеви функции на двоична информация, кодирана към фазовото пространство на n кюбитови състояния чрез елементарни n кюбитови оператори.
5. Основните градивни елементи за изграждането на общи n кюбитови оператори от виртуализираните оператори и разширяване на идеите за фазово кодиране и декодиране към тези сложни оператори.

Всички разсъждения и резултати ще предоставят солидна основа, на основата на която да се градят бъдещи изследвания в областта на квантовите изчисления.

3. Теоретични основи

Основата на квантовите изчисления не е Булева логика, а квантова логика. Към днешна дата все още няма подходящо квантово-логично пресмятане сравнимо с класическото пресмятане на Булева алгебра. Има най-малко две съществени различия между квантовата и булевата логика. Едното от тях е, че всеки квантов процес трябва да бъдат обратим, т.е. входът и изходът трябва винаги да съответстват еднозначно един на друг. По-специално,

броят на входните и изходните кюбити трябва да бъде равен, докато при Булевата логика е възможно наличието на два входни бита и само един изходен. В действителност, всички основни двоични операции на Булевата алгебра (□, □, ¬, XOR, NAND, NOR, ...) се оценяват 2-1, което означава, че те не са обратими в действителност, тъй като един □ 0 = 0 □ 1 = 0 □ 0 = 0, няма да може да се направи извод от резултата "0", който да показва стойностите на входните битове. Друга разлика между квантовата и Булевата логика е, че квантовите изходи могат да преобразуват една кюбитова основа {| 0>, | 1>}, в друга, например {| 0> + | 1>, | 0> - | 1>} , а векторната основа може да се променя от отражения или завъртания [6]. Това свойство е невъзможно в булевата логика, където всяка операция се трансформира в една от двете стойности 0 или 1. С други думи в булева логика основата никога не се променя.

4. Кюбити и гейтове

Кюбитът е квантова система, в която булеви стойности 0 и 1 са представени като определени двойки нормализирани и взаимно перпендикулярни квантови състояния означени като $\{|0\rangle, |1\rangle\}$. Двете състояния образуват „изчислителна основа“ и всяко друго (чисто) състояние на кюбита може да бъде представено като суперпозиция $\alpha|0\rangle + \beta|1\rangle$ за някои α и β така, че | α | ^{2} + | β | ^{2} = 1. Кюбитът е типична микроскопична система, като например атом, ядрен спин или поляризиран фотон. Колекция от n кюбити се нарича квантов регистър с размер п. Трябва да приемем, че информацията се съхранява в регистрите в двоична форма. Например, числото 6 е представено от регистъра със състояние $|1\rangle \otimes |1\rangle \otimes |0\rangle$. По-компактната нотация: $|a\rangle$ заема мястото на $|a_{n-1}\rangle \otimes |a_{n-2}\rangle \ldots |a_1\rangle \otimes |a_0\rangle$, където $a_i \in \{0,1\}$ и представлява квантов регистър със стойности $a = 2^0 a_0 + 2^1 a_1 + \ldots 2^{n-1} a_{n-1}$. Има 2^n състояния от този вид, които представляват всички двоични низове с дължина *n* или числа от 0 до 2^n - 1 и те образуват удобна изчислителна база. Както следва $a \in \{0,1\}^n$ (а е двоичен низ с дължина *n*), предполага се, че $|a\rangle$ принадлежи към изчислителната база. По този начин квантов регистър с размер от 3 кюбита може да съхранява три индивидуални числа, като например 3 или 7,

$$|0\rangle \otimes |1\rangle \otimes |1\rangle \equiv |011\rangle \equiv |3\rangle, \tag{1}$$

$$|1\rangle \otimes |1\rangle \otimes |1\rangle \equiv |111\rangle \equiv |7\rangle, \tag{2}$$

но регистърът може да съхранява две от тях едновременно. Защото, ако вземем първия кюбит и вместо да е $|0\rangle$ или $|1\rangle$ подготвяме суперпозиция $1/\sqrt{2}\,(|0\rangle + |1\rangle)$, след това ние получаваме:

$$\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes |1\rangle \otimes |1\rangle \equiv \frac{1}{\sqrt{2}}(|011\rangle + |111\rangle), \equiv \frac{1}{\sqrt{2}}(|3\rangle + |7\rangle). \tag{3}$$

Всъщност можем да подготвим този регистър в суперпозиция на всичките осем цифри - това е достатъчно, за да поставим всеки кюбит в суперпозиция $1/\sqrt{2}\,(|0\rangle + |1\rangle)$. Това дава

$$\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle), \tag{4}$$

което може да бъде двоично

$$|000\rangle + |001\rangle + |010\rangle + |011\rangle + |100\rangle + |101\rangle + |110\rangle + |111\rangle. \tag{5}$$

или в десетичен вид като $|0\rangle + |1\rangle + |2\rangle + |3\rangle + |4\rangle + |5\rangle + |6\rangle + |7\rangle,$ (6)

или просто като $\sum_{x=0}^{7} |x\rangle.$ Тази подготовка, както и всички други манипулации на кюбити, трябва да се извършват от унитарни операции. А квантовият гейт е устройство, което изпълнява фиксирана единна операция на избрани кюбити в определен период от време. Изходите на някои от гейтовете са свързани към входовете на другите. Размерът на мрежата представлява броят на гейтовете, които съдържа [5]. Най-често срещаният квантов гейт е този на Хадамарт [7], един хадамартов гейт извършва единична трансформация, известна като Хадамартова трансформация. Тя се определя като

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \qquad |x\rangle \boxed{H} (-1)^x |x\rangle + |1-x\rangle \tag{7}$$

Матрицата на изчислителната база $\{|0\rangle, |1\rangle\}$ и осигурява схематично представяне на гейта на Хадамарт, който действа на кюбит в състояние $|x\rangle$, $x = 0,1$.

5. Симулатор

Квантовият конструктор е Java приложение за симулиране на квантова механика, със специален акцент върху квантовите компютри. Симулаторът предлага функционалности за проектиране и изпълнение на квантови вериги. Целта на квантовият конструктор е да се даде възможност за развитие и демонстрации на квантови алгоритми. Друга основна цел е създаването на изображения и по този начин – по-добро разбиране на квантовите процеси.

Също така квантовият конструктор предлага възможност да се имплементират и изпълнят гъвкави квантови алгоритми, или квантови схеми чрез визуално-ориентирано представяне.

Въз основа на принципите на квантовата механика, квантовият конструктор осигурява прилагането на квантови регистри. Основни операции за манипулиране на регистрите, като гейта на Хадамард или C-NOT гейта са достъпни чрез лесен за използване интерфейс. Измерванията могат да се извършват на единични кюбити или върху целия размер на регистър. Освен приложението си като квантова изчислителна техника, квантовият конструктор също така е в състояние да изчисли времето на еволюцията за произволни Хамилтониани, включително на такива, които са независими от времето. Това се прави чрез числено интегриране на уравнението на Шрьодингер. Изчисляването на времето за еволюция е възможно чрез точна диагонализация на независимите от времето Хамилтонианите.

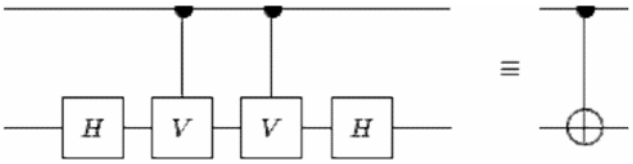
Характеристики:

- Времевата еволюция използва диагонализация от четвърти ред, включително оптимизации за откъслечни Хамилтониани.
- Възможна е имулация на произволни квантови алгоритми.
- Интегрирана декохерентност за реалистични квантови изчисления.
- Интегриран формализъм за плътностен оператор.
- Възможност за реализации на факторинг алгоритъма на Шор и алгоритъма за търсене на Гроувър[27].

6. Квантова аритметика и функции за оценяване

При квантовите компютри добавянето на всяка аритметична операция трябва да бъде вградено в единична еволюция. Ние ще се придържаме към Хадамарт и контролирано-V (C-V), и ще ги използваме като градивни елементи за всеки друг гейт и за разширителни и квантови разпространители. Ако приложим C-V четири пъти получаваме идентичност, така

че всеки три следващи приложения на C-V ще дадат обратното на C-V[8], което ще се означава C- $V^{\dagger}$. Сега, ако имаме двойка на C-V гейт и няколко гейта на Хадамарт, можем да изградим C-NOT, както следва



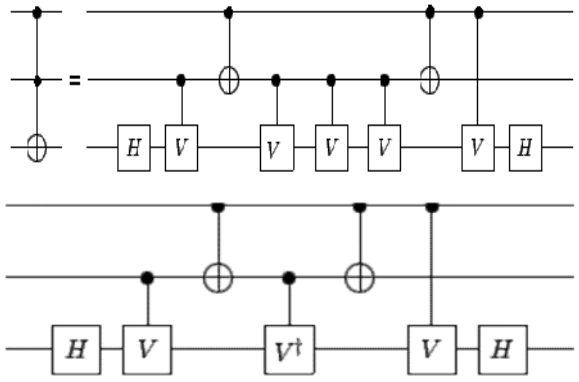
$$|0\rangle|0\rangle \to |0\rangle\frac{1}{\sqrt{2}}(|0\rangle+|1\rangle) \to |0\rangle\frac{1}{\sqrt{2}}(|0\rangle+|1\rangle) \to |0\rangle|0\rangle, \tag{8}$$

$$|0\rangle|1\rangle \to |0\rangle\frac{1}{\sqrt{2}}(|0\rangle-|1\rangle) \to |0\rangle\frac{1}{\sqrt{2}}(|0\rangle-|1\rangle) \to |0\rangle|1\rangle, \tag{9}$$

$$|1\rangle|0\rangle \to |1\rangle\frac{1}{\sqrt{2}}(|0\rangle+i|1\rangle) \to |1\rangle\frac{1}{\sqrt{2}}(|0\rangle+i^2|1\rangle) = |1\rangle\frac{1}{\sqrt{2}}(|0\rangle-|1\rangle) \to |1\rangle|1\rangle, \tag{10}$$

$$|1\rangle|1\rangle \to |1\rangle\frac{1}{\sqrt{2}}(|0\rangle-i|1\rangle) \to |1\rangle\frac{1}{\sqrt{2}}(|0\rangle-i^2|1\rangle) \to |1\rangle|0\rangle. \tag{11}$$

Една операция на кюбит не може да се извършва чрез C-NOT гейт, ако контролният кюбит е настроен на $|1\rangle$ се възприема като спомагателен кюбит. C-NOT гейтът е много по-труден за изграждане от един кюбитов NOT. Конструкцията се изчислява по следната мрежа:



$$|110\rangle \to |11\rangle\frac{1}{\sqrt{2}}(|0\rangle+|1\rangle) \to |11\rangle\frac{1}{\sqrt{2}}(|0\rangle+i|1\rangle) \to |10\rangle\frac{1}{\sqrt{2}}(|0\rangle+i|1\rangle) \to |10\rangle\frac{1}{\sqrt{2}}(|0\rangle+i|1\rangle) \to \tag{12}$$

$$\to |11\rangle\frac{1}{\sqrt{2}}(|0\rangle+|1\rangle) \to |11\rangle\frac{1}{\sqrt{2}}(|0\rangle-i^2|1\rangle) = |11\rangle\frac{1}{\sqrt{2}}(|0\rangle+|1\rangle) \to |111\rangle. \tag{13}$$

$$|111\rangle \to |11\rangle\frac{1}{\sqrt{2}}(|0\rangle-|1\rangle) \to |11\rangle\frac{1}{\sqrt{2}}(|0\rangle-i|1\rangle) \to |10\rangle\frac{1}{\sqrt{2}}(|0\rangle-i|1\rangle) \to |10\rangle\frac{1}{\sqrt{2}}(|0\rangle-i|1\rangle) \to \tag{14}$$

$$\to |11\rangle\frac{1}{\sqrt{2}}(|0\rangle-i|1\rangle) \to |11\rangle\frac{1}{\sqrt{2}}(|0\rangle-i^2|1\rangle) = |11\rangle\frac{1}{\sqrt{2}}(|0\rangle+|1\rangle) \to |110\rangle. \tag{15}$$

Този гейт е с два контролни кюбита (първите две линии на диаграмата) и един кюбит, който е обратен, само когато двете контроли са в състояние $|1\rangle|1\rangle$. c^2-NOT гейтът е логическото съединение, от което имаме нужда за аритметиката. Ако целта първоначално се

установява при $|0\rangle$, действието е обратимо и гейт след гейт операцията става логична и за два контролни кюбита.

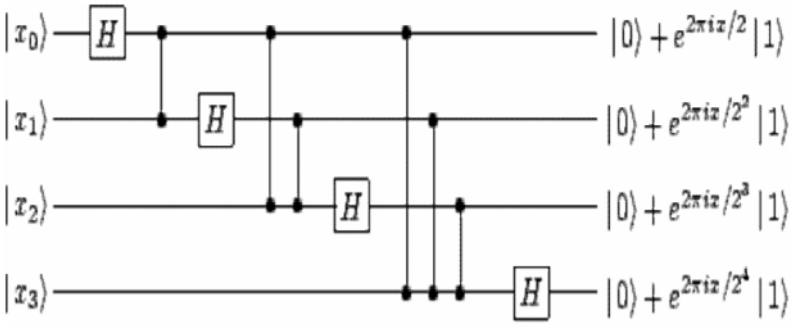
$$|x_1, x_2\rangle|0\rangle \mapsto |x_1, x_2\rangle|x_1 \wedge x_2\rangle \tag{16}$$

7. Алгоритми и тяхната сложност

За да се реши даден проблем, независимо дали компютрите са класически или квантови, те следват точен набор от инструкции, които могат да се прилагат автоматично, за да се получи решение за всеки един момент от процеса. Спецификацията на този набор от инструкции се нарича алгоритъм. Всеки алгоритъм може да бъде представен като булева мрежа (N_1, N_2, N_3, ...), където мрежата N_n действа за всички възможни случаи на вход с размер n бита. Имаме нужда от един алгоритъм, машина на Тюринг, който да преобразува всеки n в явно описание на N. Размерът на квантовото преобразуване на Хадамарт се увеличава с N, с увеличаването на броя на входните кюбити. Друг добър пример от сферата на мрежите е квантовата трансформация на Фурие определена в изчислителната база като унитарна операция.

$$|y\rangle \mapsto 2^{-n/2} \sum_x e^{i\frac{2\pi}{2^n}yx}|x\rangle, \tag{17}$$

Да предположим, че искаме да изградим такава унитарна еволюция на N кюбити . Можем да започнем с един кюбит, в този случай трансформацията на Фурие се свежда до прилагане на гейта на Хадамарт. След това можем да вземем два кюбита и трансформацията на Фурие може да се реализира с два Хадамартови гейта и контролирана променлива фаза B (π) помежду им. По този начин можем да изградим три и четири кюбита с трансформацията на Фурие, чиято мрежа изглежда така:



$$H\ B(\pi)\ H\ B(\pi/2)B(\pi)\ H\ B(\pi/4)B(\pi/2)B(\pi)\ H$$

Забележка: има три различни вида на B (φ) гейтове в мрежата по-горе: B (π), B (π/2) и B (π/4).

Големият проблем при проектирането на алгоритмите или съответните им семейства от мрежи е оптималното използване на физическите ресурси, необходими за решаване на проблема. Сложността на теорията идва от факта, че тя се занимава с необходимия разход на изчисления по отношение на някои определени елементарни операции, използването на паметта или размера на мрежа. Алгоритъмът е бърз и ефективен, ако броят на елементарните операции за изпълнение не се увеличава по-бързо от полиномиалната функция за размера на входа.

8. Изводи и заключение

Конструкторът за квантови изчисления установява обобщение на примитивните квантови оператори, подходящи за употреба при проектиране и анализ на квантови

изчислителни вериги. Освен това той позволява подход към квантовите изчисления, при който всички оператори кодират дискретна, двоична информация относно състоянията, върху които действат във фазата на резултата от оператора. Такива кодирания могат да бъдат използвани след това като основа за генериране на модели на изчисления, ефективно декодиращи информация, която преди това е била кодирана във фазово пространство. Логиката в основата на квантовия конструктор също предоставя няколко нови средства за обобщаване на основни n кюбитови оператори и класифициране на операторите за n кюбитови вериги.

Литература

[1] Goldreich O. and Vadhan S. P. (1992). Communication via one- and two- particle operators on Einstein-Podolsky-Rosen states, 69(20):2881–2884. Physical Review Letters.
[2] David P. DiVincenzo. Quantum computation. Science, 270:255–261, 1995.
[3] Mayfield, J. L., and Ralescu, A. L. Universal amplitude/phase gates for quantum algorithm design. In Proceedings of the Seventeeth Midwest Artificial Intelligence and Cognitive Science Conference (2006), MAICS, pp. 79–84.
[4] Charles H. Bennett, Ethan Bernstein, Gilles Brassard, and Umesh V. Vazirani. Strengths and weaknesses of quantum computing. SIAM Journal on Computing, 26:1510–1523, 1997.
[5] Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Michael Sipser. Quantum computation by adiabatic evolution. Technical Report MIT-CTP-2936, MIT, Jan. 2000.
[6] Wim van Dam, Michele Mosca, and Umesh Vazirani. How powerful is adiabatic quantum computation? In Proc. of the 42nd Annual Symposium on Foundations of Computer Science (FOCS2001), pages 279–287. IEEE, 2001.
[7] Gilles Brassard, Peter Hoyer, and Alain Tapp. Quantum counting. In K. Larsen, editor, Proc. of 25th Intl. Colloquium on Automata, Languages, and Programming (ICALP98), pages 820–831, Berlin, 1998. Springer. Los Alamos preprint quant-ph/9805082.
[8] Sebastian M. Maurer, Tad Hogg, and Bernardo A. Huberman. Portfolios of quantum algorithms. Physical Review Letters, 87:257901, 2001. Los Alamos preprint quant-ph/0105071.

За контакти:
Николай Райчев,
Факултет по изчислителна техника и автоматизация,
Технически университет - Варна,
Варна 9010, ул."Студентска" № 1, България
Телефон: +359-0889-852720
Email: afron@abv.bg

РАЗВИТИЕ НА GPU И ПОВИШАВАНЕ НА ПРОИЗВОДИТЕЛНОСТТА С CUDA

Венцислав Петров, Емил Френски, Димитър Манолев

Резюме: Статията съдържа обобщение на еволюцията на GPU архитектурата - от специфично ядро, фиксирана функция на хардуера, направен за изпълнение единствено на графики, до голям набор от паралелни и програмируеми ядра за общи компютърни изчисления. В статията се дава и обща представа за най-важните научни области, в които GPU могат да бъдат използвани за изчисления с общо предназначение. Накрая сме избрали и компютърна конфигурация за нашите експерименти.
Ключови думи: Изчисления с общо предназначение с GPU, CUDA на NVIDIA.

Development of GPU, implementation of GPU for parallel processing Performance with CUDA by NVIDIA

Ventsislav Petrov, Emil Frenski, Dimitar Manolev

Summary: The article contains a summary of the evolution of GPU architecture - from specific core fixed function of the hardware designed to perform only graphics to a wide range of parallel and programmable cores for general computing. The article gives an overview of the most important research areas in which the GPU can be used for general purpose computing. Finally, we have chosen a computer configuration for our experiments.
Keywords: General purpose computing with GPU, NVIDIA's CUDA.

1. Увод

Развитието на графичния хардуер позволява съществено изменение на начина, по който тези системи могат да се ползват за общо приложими математически изчислителни задачи. Тези разработки довеждат до развитието на концепцията за използване на графичния хардуер за компютърни изчисления с общо предназначение - GPGPU (General Purpose Computation Using Graphics Hardware). Тази концепция позволява бързото решаване на множество сложни последователни алгоритми подлежащи на паралелизация, каквито са обработката и филтрацията на сигналите, компресирането на видео, криптографски анализ, статистически изчисления и други алгоритми обработващи еднотипно големи масиви от данни.

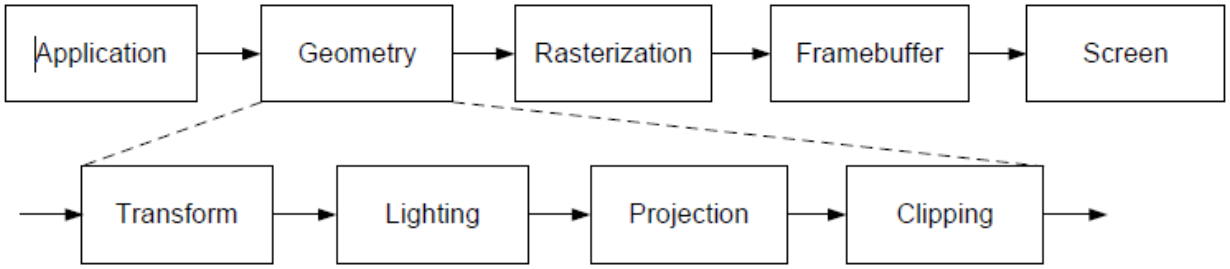
Графичният процесор (GPU) е отделен процесор, оптимизиран за ускоряване на графичните изчисления. GPU е проектиран специално за паралелно изпълнение на много изчисления с плаваща запетая, което е от съществено значение особено за 3D графичното рендиране. Съвременните графични процесори са масивно паралелни и напълно програмируеми. Паралелните изчисления на действия с плаваща запетая в съвременните GPU са от порядъци по-високи от същите извършени от CPU [6].

Първоначално графичните процесори са моделирани, следвайки идеята за графичния канал. Той е концептуален модел от етапи, през които графичните данни преминават и обикновено се изпълнява чрез комбинация от хардуер (GPU ядра) и CPU софтуер (OpenGL, DirectX). Графичният канал трансформира координати от 3D пространството (уточнени от програмиста) в 2D пространство от пиксели на екрана. Графичният канал всъщност е "поточна линия" от различните етапи на операциите, които се прилагат за триъгълниците/пиксели и могат да бъдат обобщени в два етапа: Геометрия и Рендъринг. В началото

графични процесори извършват само Рендъринг етапа в хардуера, а CPU се използва за генериране на триъгълници, за да могат GPU да ги обработват. Тъй като GPU технологията се развива, все повече и повече етапи от графичния канал се изпълняват от GPU, освобождавайки повече CPU [6].

Графичният канал (наричан също и рендериращ канал) е модел, който описва различните стъпки, необходими, за да обработи една сцена. Концепцията за графичния канал може да бъде сравнена с инструкционния канал при CPU. Отделните стъпки се извършват паралелно, но са блокирани докато и последната стъпка не свърши. Един опростен модел на (фиксирана функция) графичен канал е изобразен на фиг. 1.

Приложението променя сцената, например реагира на потребителския вход. Компонентите на новата сцена се препращат за трансформация. При тази стъпка локалните координати на обектите се трансформират в глобална координатна система. Камерата е разположена в сцената и координатната система се коригира към нея. По време на осветяването



Фиг. 1: Опростен модел на графичен канал

се изчисляват стойностите на цветовете за всички върхове, в зависимост от позицията на светлината и свойствата на съответстващите триъгълници. По време на стъпката на проекцията 3D сцената се нанася на 2D пространството на картината. През клипинга, ненужните примитиви се елиминират. През стъпката за растеризация скритите повърхности се премахват (използвайки z-buffer алгоритъм) и сцената се превръща в растерна графика чрез изчисляване стойности за цвета на всеки пиксел.

2. Поколения на графичните карти

2.1. Поколение 0

GPU хардуерът и графичният канал започват наистина да се оформят през 1993 г, когато SGI пуска своята Reality Engine платка за графична обработка [5]. Имало е отделни платки и чипове за по-късните фази от графичния канал, но все пак се разчитало на CPU през първата му половина. Данните са с фиксиран поток през всеки един етап. Комбинация от „евтин" хардуер с игри като Quake и Doom ускорява развитието на „игралната" индустрия и възприемането на GPU [4]. Дори и с тези графични канали обаче, първите графични процесори все още извеждат само един пиксел на такт, което означава, че CPU изпраща повече триъгълници към GPU, отколкото той може да обработи. Това е довело до добавянето на повече графични канали работещи паралелно (и в крайна сметка до повече графични ядра), така че да се обработват паралелно множество пиксели на всеки един такт [4].

2.2. Поколение I

Картата 3dfx Voodoo (1996) е считана за една от първите истински 3D „геймърски" графични карти [9]. Само тя предлага 3D ускорител, но ние все още се нуждаем и от 2D ускорител. Тя работи на PCI шината. CPU все още извършва трансформациите на върховете,

докато Voodoo осигурява етапите: текстурното разпределяне (texture mapping), z-buffering, и растеризацията (rasterization).

2.3. Поколение II

През 1999 г, са реализирани вече първите карти за изпълнението на фазите в целия графичен канал (сега вече и с изчисления за трансформация и осветление) на GPU. С появата на GeForce256 NVIDIA и ATI Radeon 7500, на разположение на потребителя са първите истински графични процесори.

До 1999 г., терминът "GPU" всъщност не съществува. NVIDIA въвежда термина по време на стартирането на видеокартата GeForce 256 [4]. Това поколение карти е първото, което използва новия графичен порт (AGP) вместо на шината PCI, и предлага нови графични функции в хардуера, като например мулти текстуриране, карти на осветеността и хардуерна трансформация на геометрията и осветлението [4]. Първите графични канали са били известни като канали с „фиксирана функция", защото след като програмистът изпрати графичните данни към графичния канал на GPU, данните не може да се променят [4]. Въпреки че така обработката е много по-бърза, основният проблем с фиксиран модел функция на канала е липсата на гъвкавост за графични ефекти.

2.4. Поколение III

Следващата стъпка в еволюцията на GPU хардуера е въвеждането на програмируемите графични канали на GPU. През 2001, NVIDIA пуска Ge Force 3, която дава на програмистите възможността да правят програми за частите от графичния канал, които са били с „фиксирана функция" [5]. Вместо да се изпратят всичките данни към GPU и те просто да преминат през фиксирания графичен канал, програмистът сега може да изпрати тези данни заедно с вертекс (vertex) "програми" (наречени shaders), които обработват данните, докато са в графичния канал [5]. Тези shader програми са малки ядра - "kernels", написани на shader езици, подобни на асемблер.

2.5. Поколение IV

Една година по-късно, през 2002, на пазара вече са първите напълно програмируеми графични карти: NVIDIA GeForce FX и ATI Radeon 9700. Тези карти позволяват операции върху пиксели с програмируеми вертексни и пиксел (фрагмент) шейдъри, и позволяват за ограничена употреба специфично „свързване" mapping на входно-изходни операции с данни [5].

През 2003 GPU започва да се използва за компютърни изчисления (приблизително с излизането на DirectX 9) за не графични данни. Започват да се появяват и поддръжка на плаваща запетая и разширена текстурна обработка в картите.

2.6. Поколение V

През 2004 г. GeForce 6 и Radeon X800 излизат на пазара и са едни от първите карти за използване на PCI-Express шината.

Относно софтуера - започват да се появяват GPU езици от високо ниво като Брук и Sh, които предлагат истински условности и цикли както и динамичен контрол на потока в шейдърните програми.

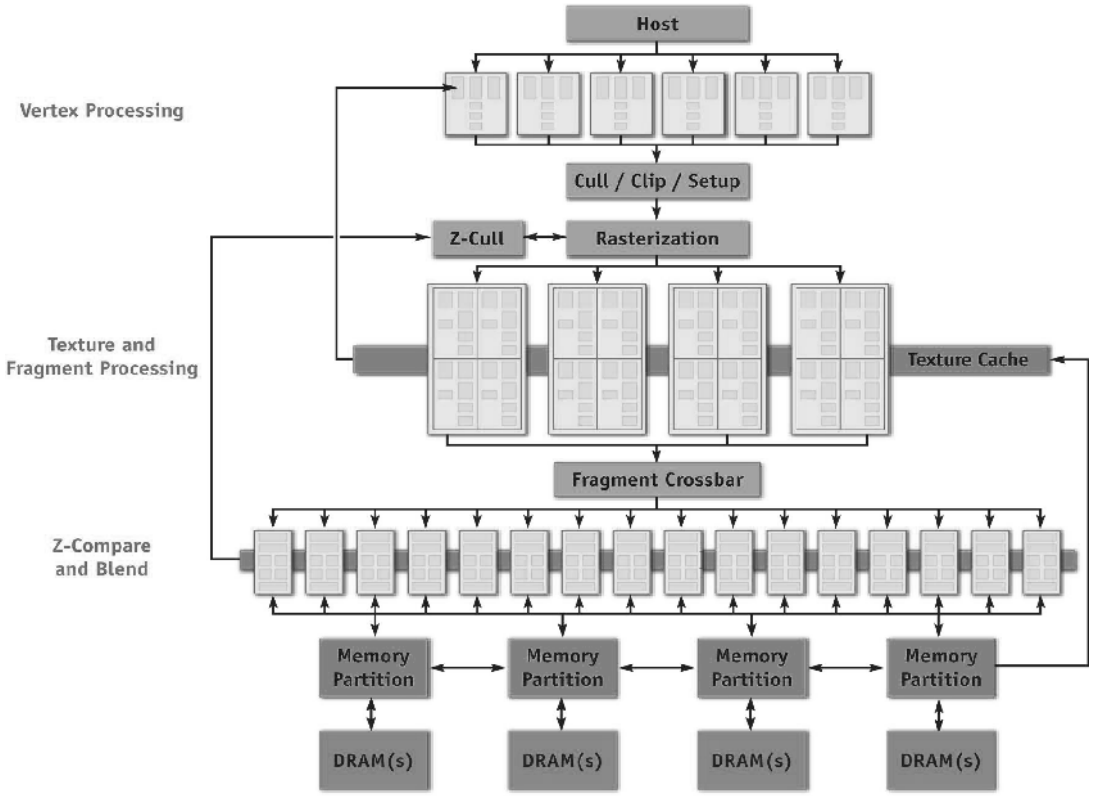
От страна на хардуера се въвеждат по-висока точност (64-битова двойна поддръжка), много буфери за рендиране, увеличена GPU памет и получава достъп до текстурите (фиг. 2). [5].

Когато се разглежда графичният канал за обработка на графика, то той съдържа програмируемо vertex устройство, програмируемо фрагмент устройство, текстурно устройство и устройство за дълбочина - сравнение/смесване - запис на данни. Когато се разглежда канала като алтернатива като CPU (за не графични приложения), на GPU може да се гледа като на голямо количество програмируеми мощности с плаваща запетая и широколентова памет, която може да се използва на интензивни изчисления с приложения, които не са свързани с компютърната графика. Започва тенденция към програмируемост на GPU.

2.7. Поколение VI

Появата на NVIDIA's GeForce 8 серията видеокарти през 2006 бележи следващата стъпка в еволюцията на GPU – представяне на GPU като масивно паралелни процесори [5]. Архитектурата на G80 (GeForce 8800) е първата, която има "унифицирани", програмируеми shaders – т.е. напълно програмируем унифициран процесор наречен Streaming Multiprocessor (поточен мултипроцесор) или SM, който управлява вертексните, пикселните и геометричните изчисления. Въведен е нов геометричен шейдър, добавяйки повече програмируемост при комбиниране на вертекс шейдър и пиксел шейдъри. С този унифициран дизайн на шейдърите, традиционния модел на графичния канал вече е чисто софтуерна абстракция.

За да използва това качество на графичните процесори се въвежда новият език за програмиране CUDA от NVIDIA. Не много по-късно се създава и ATI Stream за ATI карти и DirectX 10 за всички карти (за Microsoft Windows само) [6].



Фиг. 2. GeForce 6 архитектура

2.8. Поколение VII

Тенденцията към повече CPU-подобни, програмируеми GPU ядра продължава с въвеждането на NVIDIA's Fermi архитектурата. Анонсирана в края на 2009, но реализирана в

началото на 2010, Fermi видеокартите са първите GPU проектирани за GPGPU (General purpose graphics processing unit) изчисления, внасяйки функции като: истинска йерархия на кеша, ECC, унифицирано адресно пространство на паметта, едновременни изпълнения на ядрото (kernel), по-добра производителност при изчисленията с двойна точност [3]. Видеокартата GTX480 Fermi има общо 480 CUDA ядра (15 поточни мултипроцесора - SM с 32 CUDA ядра всеки) [3].

2.9. Поколение VIII

NVIDIA® Kepler - най-бързата и по-ефикасната високопроизводителна компютърна архитектура в света (HPC). С новите компютърни технологии и характеристики тя е приложима за широк кръг от научни изчислителни приложения и прави хибридните изчисления по-достъпни за разработчиците на приложения и изследователите.

SMX - Доставя повече производителност и ефективност чрез новия си и иновативен поточен мултипроцесорен дизайн, което позволява голям процент от пространството да се използва от процесорните ядра в сравнение с управляващата логика.

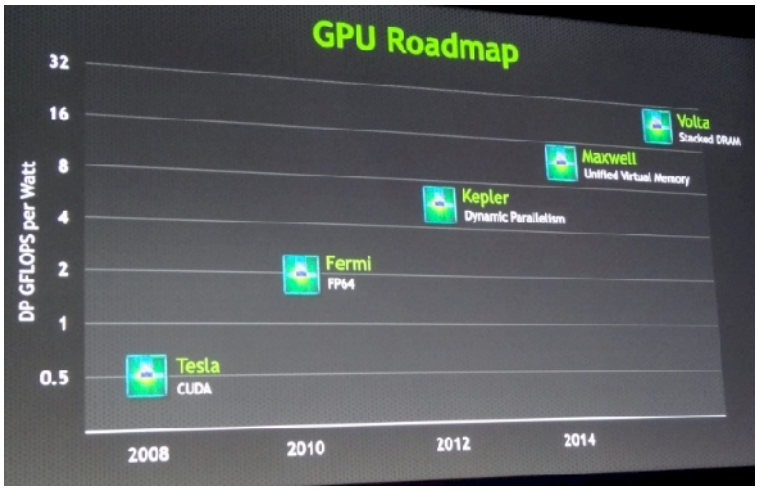
Динамичен паралелизъм - Опростява GPU програмирането, като позволява на програмистите по-лесно да ускорят всички паралелни вложени цикъла, в резултат на което GPU динамично размножава нови нишки в себе си, без да се обръща към CPU.

Hyper-Q - Позволява няколко ядра на процесора едновременно да използват един Kepler GPU, което драстично увеличава програмируемостта и ефективността.

2.10. Следващи поколения

През март 2013 Nvidia анонсира наследника на архитектурата Kepler, а именно Maxwell архитектурата. Планира се тя да се появи през 2014 [8]. Пак през март 2013 Nvidia анонсира че като наследник на Maxwell ще бъде Volta архитектурата [8].

Maxwell архитектурата, наследника на Kepler, ще има за пръв път интегриран ARM процесор (Project Denver). Това ще направи Maxwell GPU по-независим от основния CPU. Относно Volta архитектурата, Nvidia единствено разкрива нейното име и споменава, че тя ще има подредена DRAM [9]. На фиг. 4 е дадена картата за развитието на GPU Nvidia.



Фиг. 3. GPU Roadmap

3. Области на приложение и софтуер базиран на CUDA

По-долу са изброени част от областите, където процесите могат да бъдат ускорени, използвайки CUDA паралелните изчисления на NVIDIA видеокартите:

- Правителство & отбрана;
- Молекулярна динамика, изчислителна химия;
- Био-информатиката;
- Електродинамика и електромагнетизъм;
- Медицински изображения, CT, MRI;
- Добив на нефт и газ;
- Финансови изчисления и ценообразуване;
- MATLAB, LABVIEW, Mathematica, R;
- Автоматизация на проектирането в електрониката;
- Прогнозиране на времето и моделиране океана и пренос на замърсявания.

Някои от най-популярните приложения, работещи много по-бързо, използвайки CUDA, са:

- GPUGRID.net е Novel дистрибутирана суперкомпютърна инфраструктура, направена от много NVIDIA графични карти съединени заедно, за да дадат висока производителност на всички биомолекулярни симулации;
- NVIDIA и MathWorks си сътрудничат, за да дадат мощност, осигурена от GPU изчисленията за MATLAB потребителите. Това е налично от версията на MATLAB 2010b. NVIDIA GPU ускорението позволява да се постигат по-бързо резултатите за потребителите на Parallel Computing Toolbox и MATLAB Distributed Computing Server;
- LabVIEW GPU изчисленията разгръщат изчислителната мощ на NVIDIA графичните процесори при използването на LabVIEW. Интегриран е код, който извиква GPU за изчисления в собствената паралелна система на LabVIEW.

4. Компютърна конфигурация за GPGPU

Избрахме следните компоненти за компютърна конфигурация, подходяща да ускори изчисленията при задачи подлежащи на паралелна обработка с NVidia и CUDA:

Mainboard - GIGABYTE GA-Z87X-UD3H;

CPU - Intel® Core™ i5-4430 (6M Cache, up to 3.20 GHz)

Video card – PALIT GeForce® GTX 650 Ti BOOST OC (2048MB GDDR5)

memory interface 192bit, Graphics Clock (MHz) Base Clock 1006MHz/Boost Clock 1072MHz, memory clock 3054MHz (DDR 6108MHz), CUDA cores 768, memory bandwidth 144.2GB/sec;

RAM - DDR3 4Gb, 1333MHz, A-Data – 2 бр, работещи в Dual Chan-nel режим - общо 8Gb;

PSU - CFT-750-14CS CHIEFTEC (По-мощно ще бъде по-добре, но има ограничение в средствата).

Останалите компоненти не са съществени за продуктивността на системата.

5. Заключение

Еволюцията на GPU хардуера до този момент преминава от силно специфично единично ядро, с фиксирани функции графичен канал с приложение само за графично рендиране, до набор от масивно паралелни и програмируеми ядра за повече изчисления с общо предназначение. Сега архитектурата на многоядрените графични карти изглеждат все повече и повече като многоядрените процесори, използвани за изчисления с общо

предназначение. В този аспект Kepler архитектурата може да се разглежда като система от 15 процесора, всеки от които с по 192 ядра.

Литература

[1] Buck, Ian. The Evolution of GPUs for General Purpose Computing. GTC 2010;
[2] Crow, Thomas. Evolution of the Graphical Processing Unit. 2004 Paper;
[3] Fermi Whitepaper. 2010 NVIDIA;
[4] GPGPU website. http://www.gpgpu.org/;
[5] Luebke, David. GPU Architecture: Implications & Trends. SIG-GRAPH 2008;
[6] Video Card. http://en.wikipedia.org/wiki/Video_card.
[7] NVIDIA Kepler GK110 next generation CUDA compute architec-ture Whitepaper 2013 NVIDIA;
[8] Smith, Ryan (March 19, 2013). "NVIDIA Updates GPU Roadmap; Announces Volta Family For Beyond 2014". An-andTech. Retrieved March 19, 2013;
[9] Nvidia GTC 2013.

За контакти:
гл. ас. инж. Венцислав В. Петров
катедра „Компютърни системи и технологии”
ЮЗУ „Неофит Рилски”
E-mail: ventsislav_p@swu.bg

АНАЛИЗ НА СИГУРНОСТТА В GSM КОМУНИКАЦИИТЕ

Антония Т. Ташева, Жанета Н. Ташева

Резюме: В статията е направен анализ на основните атаки на по-силния GSM шифър A5/1. Разгледани са първите реални хардуерни реализации на атаки, представени в общодостъпната литература. В резултат на анализа е направен извод за необходимостта от допълнително криптиране на информацията в GSM комуникациите когато се предават чувствителни и конфиденциални данни.
Ключови думи: Криптоанализ, GSM сигурност, криптиране.

Security analysis of GSM communications

Antoniya T. Tasheva, Zhaneta N. Tasheva

Abstract: GSM (Group Special Mobile) is the most widely used telecommunication system. Analysis of the security of the stronger GSM cipher A5/1 is made in this paper. The first real FPGA hardware-based attacks against A5/1 are examined and analyzed. A conclusion is made about the necessity of additional encryption of sensitive and confidential information sent through GSM networks.
Keywords: Cryptanalysis, GSM Security, Encryption

1. Увод

Глобалната система за мобилни комуникации GSM (Global System for Mobile communications, първоначално Group Special Mobile, 1982 г.) е най-широко използваната система за мобилни телефонни комуникации. Смята се, че повече от 3 милиарда абонати в целия свят използват мобилни телефони, базирани на тази система. GSM стандартът дефинира алгоритъм за удостоверяване, както и за криптиране на данните. Оригиналният криптиращ шифър A5/1 се използва в Европа. С цел разгръщане на GSM извън Европа и удовлетворяване на изискването за ограничение на износа, по-късно умишлено е разработен алгоритъма A5/2 като по-слаб шифър. Първоначално дизайнът и на двата шифъра е бил държан в тайна, но общата идея е изтекла от Андерсън в доклад през 1994 г. Дизайнът на двата шифъра A5/1 и A5/2 е изцяло представен от Бричено в 1999 [5]. Веднага след това слабият шифър A5/2 е криптоанализиран от Голдбърг, Вагнер и Грийн.

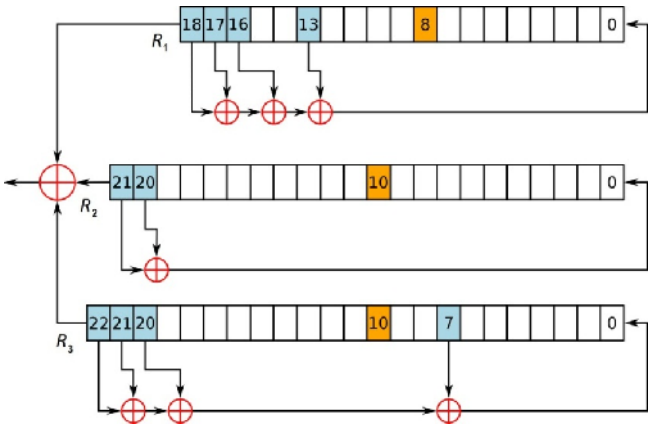
В статията е направен анализ на сигурността на по-силния GSM шифър A5/1. Шифърът е широко анализиран още с изтичането на информация за неговия дизайн [1, 4, 3, 8, 11]. Много от предложените първоначални атаки срещу A5/1 никога не са били напълно приложени и/или им липсва практичност при конкретната реализация. Едва в последните години с масовото навлизане на евтини чипове с програмируема логика FPGA реалното изпълнение на GSM криптоатаките е възможно.

2. Шифър A5/1

A5/1 е синхронен поточен шифър. Генерираният от A5/1 ключов поток побитово се сумира с откритият текст, при което се получава криптотекст. Комуникацията между мобилния телефон и базовата станция BS (Base Station) е разделен на кадри, всеки от по 114 бита. За всеки кадър се генерира нов ключов поток. Всички кадри на един телефонен разговор имат един и същ ключ на сесията $K = (k_0, k_1, \ldots, k_{63}) \in \mathrm{GF}(2^{64})$, който се обменя между мобилния телефон и BS в началото при установяване на разговора. 22-битов инициализиращ вектор $\mathrm{IV} = (v_0, v_1, \ldots, v_{21}) \in \mathrm{GF}(2^{22})$ е уникален за всеки кадър от

телефонния разговор, но IV е равен на 22-битовия публично известен номер на кадъра FN (Frame Number).

Шифърът A5/1 се състои от три преместващи регистъра с линейни обратни връзки LFSR (Linear Feedback Shift Registers) R_1, R_2 и R_3 с дължина съответно 19, 22 и 23 бита (фиг. 1). Обратните връзки на всеки регистър отговарят на коефициентите на примитивни полиноми от степен, еднаква с дължините на регистрите. Следователно всеки LFSR генерира последователност с максимален период на повторение. Най-старшите битове на всички 3 регистъра се сумират по модул 2 и генерират един бит от ключовата последователност.



Фиг. 1. Схема на шифър A5/1

Регистрите се тактуват асинхронно, като тактовите сигнали на всеки регистър C_1, C_2 и C_3 се задават с равенствата:

$$C_1 = R_1[8],\ C_2 = R_2[10] \text{ и } C_3 = R_3[10]. \quad (1)$$

На всеки такт се изчислява битът M като функция на трите тактови сигнала:

$$M = C_1C_2 + C_1C_3 + C_2C_3. \quad (2)$$

Регистърът R_i, $i = 1, 2, 3$, се тактува, ако е $C_i = M$, в противен случай той се спира. На всеки такт се тактуват 2 или 3 регистъра, следователно вероятността за тактуване на даден регистър е равна на 3/4.

Алгоритъмът за генериране на ключовия поток е следният: В началото трите регистъра R_1, R_2 и R_3 се нулират. Следва *фаза на инициализация*, по време на която те се тактуват синхронно. За 64 такта регистрите се инициализират с 64-битов ключ K: на i-тия такт, $i = 0, 1, 63$, i-тият бит от ключа се сумира по модул 2 с най-младшия бит на всеки регистър и регистрите се изместват наляво. Ключът K е последван от 22-битов инициализиращ вектор IV, който отново се сумира по модул две с младшите разряди.

След инициализиращата фаза следва *предварителна фаза*, по време на която регистрите се тактуват асинхронно. Шифърът, който се генерира за 100 такта, се отхвърля и не се използва. Следва *същинската фаза*, в която се генерират 228 бита ключов поток, 114 от които се използват за криптиране на изходящия трафик и 114 – за декриптиране на входящия трафик. Процесите се повторят за следващия кадър.

3. Основни атаки на шифъра A5/1

Докато криптоанализът на блоковите шифри се фокусира върху намиране на използвания секретен ключ, то криптоанализът на поточните шифри основно е насочен към намиране на вътрешното състояние на шифъра [4, 8]. След това шифърът може да продължи своята работа и да декодира остатъка от криптирания текст.

Атака на грубата сила. Един от подходите за разбиване на шифъра е да се криптира известен текст P с всички възможни стойности на ключа K, докато резултатът съвпадне с шифротекста C. Този подход е приложим само за шифри с относително малки ключови размери, тъй като атаката на грубата сила изисква дълго време за изчисляване $T = N$.

Таблица за претърсване. Възможно е да се избере определен текст P_0, който вероятно ще се появи в криптираните данни. Такъв открит текст може да бъде, например, заглавната част на файл, последователността от интервали в текстов файл и др. За избран открит текст P_0 се изчисляват всички двойки ключ-шифротекст (K_i,C_i) и се съхраняват в паметта, сортирани по шифротекста. При извършване на действителната атака, шифротекстът се търси в записаната таблица и ключът се възстановява. Фазата на предварително изчисляване (наречена офлайн фаза) изисква висока мощност на изчисляване, но този етап се изпълнява само веднъж. Самата атака (наречена онлайн фаза) след това е много бърза, $T = 1$. Този подход обаче, изисква голям обем данни и памет, като сложността е $M = N$.

Двата метода за атаки са алтернативни, като първият изисква максимално време T, а вторият – максимален обем памет M, съответно равни на сложността на задачата N.

Методът за атака чрез компромис между време и памет TMTO (Time-Memory Trade-Off) е въведен от Мартин Хелман през 1980 г. [10]. Целта на метода е да се намали времето, необходимо за разбиване на шифъра, което е особено важно, ако атаката се повтаря често. По време на офлайн фазата се изчисляват m на брой вериги, като в таблицата се запомнят само двойките от началния и краен ключ, сортирани по крайния. По време на онлайн фазата по даден криптотекст се цели намиране на ключа K, като се прилага случайната функция R към шифротекста и резултатът се сравнява с крайните точки в таблиците. Ако се намери съвпадение в ред на таблицата, то се определя ключът K или крайната точка има множество инверсни образи, т.е. тя е фалшива точка.

Методът първоначално е разработен за блоковия шифър DES, като след това са предложени някои негови подобрения и изменения. За да намали броят на достъпите до паметта по време на атаката, Ривест [6] предлага модификация на основата на т. нар. **отличими точки DP** (Distinguished Points). DP-точките са ключове, които могат лесно да се проверят, например, ключ с нулеви 20 значещи бита. DP-точките обикновено се характеризират с маска с дължина d бита. По време на офлайн фазата се генерират вериги до получаването на отличима точка. В следствие на това веригите нямат константна дължина.

Бирюков и Самир [4] комбинират метода на Хелман с атаката на поточните шифри, разработен от Бабидж [1] и Голик [8]. Резултат от тази комбинация е методът на **компромис между време, памет и данни** TMDTO (Time-Memory-Data Trade-Off), приложим особено за поточните шифри. Основната идея на метода е следната. Нека g е функция на съответствие на всяко от N-те начални състояния $x \in \boldsymbol{X}$ на поточния шифър в изходното $y \in \boldsymbol{Y}$:

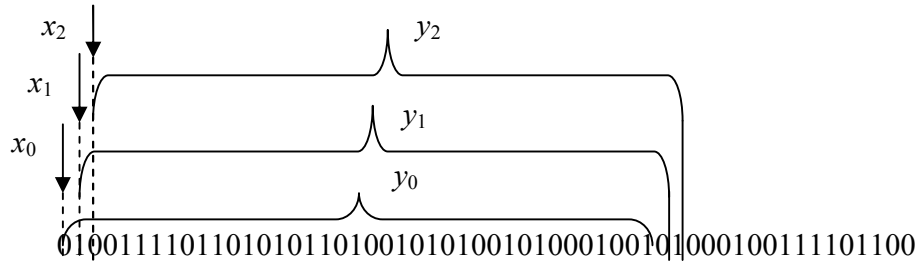
$$y = g(x),\ g : \boldsymbol{X} \rightarrow \boldsymbol{Y}. \tag{3}$$

Целта на криптоаналитика е да инвертира функцията g, т.е. за известен изход y да намери такова x, което да удовлетворява горното уравнение (3). Ако n е дължината на прихванатите изходни битове и $n > \log_2 N$, то могат да се запишат

$$D = n - \log_2 N + 1 \tag{4}$$

изходни последователности $y_0, y_1, ..., y_{D-1}$, както е показано на фиг. 2.

За да се разбие поточният шифър, е достатъчно да се намерят вътрешните състояния x_i за всички D на брой изходи y_i. След това шифърът може да се пусне да работи напред и да генерира следващите изходни последователности.



Фиг. 2. Запис на D на брой изходни последователности от прихванатите битове

Методът с **таблици дъга** (rainbow tables) [13] е най-добрият в момента за криптоанализ на блоковите шифри, но е по-лош подход от оригиналния на Хелман в случай на поточни шифри. Съхраняването на множество различни точки в таблиците води до намаляване на времето за предварително изчисляване на таблиците и използваната памет, но заедно с това нараства времето за онлайн фазата и достъпа до таблиците. За да премахнат тези недостатъци през 2006 г. Баркам, Нихам и Шамир [2] представят модифициран метод на таблиците дъга, наречен **таблици тънка дъга** (Thin-Rainbow Tables), който е ефикасен, дори и в случай на множество входни данни. При него по-къси последователности дъги от S на брой различни функции се повтарят на всяка стъпка t пъти във всяка верига:

$$f_1, f_2, \ldots, f_S, f_1, f_2, \ldots, f_S, \ldots, f_1, f_2, \ldots, f_S. \quad (5)$$

Така всяка верига ще съдържа $tS+1$ точки и ако $t \geq D$, може да се постигне максимален компромис при налични множество данни. При генериране на множество таблици, всяка от тях има своя уникална последователност дъга $f_1, f_2, \ldots, f_S$.

4. Хардуерни реализации на атаки на шифъра A5/1

Криптоанализът на модерните криптографски алгоритми изисква висока мощност на изчислителния процес, която може да се предостави от суперкомпютри, като IBM BlueGene, Cray или SGI; разпределени изчислителни системи; интегрални схеми със специални приложения ASICs (Application Specific Integrated Circuits) и масиви с програмируеми полета FPGAs (Field-Programmable Gate Arrays).

С подобренията в FPGA технологията преконфигурируемите изчисления се налагат като ефективна, евтина алтернатива за някои приложения на суперкомпютрите. За разлика от стандартните процесори преконфигурируемите вериги предоставят много по-висока изчислителна производителност на сравнително по-ниска цена. Те не достигат производителността на специалните ASICs, обаче тяхното препрограмиране дава възможност за използването им за различни проекти. Еднократните разходи за проектиране са много по-ниски отколкото да се проектират специални ASIC чипове. Поради това, изработването на устройства за криптоанализ със специално предназначение вече е достъпно и за институции извън правителствените агенции.

Паралелното устройство за разбиване на кодове с оптимизирана цена COPACOBANA е високо производителен клъстер, състоящ се от 120 Xilinx Spartan3 XC3S1000 FPGAs, разработен съвместно от Университет „Кристиян Албрехтс“ в Кил и Рурски Университет в Бохум през 2006 [12]. В момента COPACOBANA е преконфигурируема паралелна FPGA машина, оптимизирана за задачи по разбиване на кодове. В зависимост от алгоритъма, паралелната архитектура на хардуера може да надмине по производителност конвенционалните компютри с няколко порядъка. COPACOBANA е проектирана за изпълнение на типичната криптоаналитична задача, която се характеризира с паралелни операции, които имат много ограничена необходимост от комуникация една с друга, малък трансфер на данни между хоста и възлите, защото изчисленията доминират над

комуникациите и се изисква много малко локална памет, която може да бъде реализирана от RAM модулите на FPGA.

Атаката „предположи и определи“. Първата реализирана хардуерна атака с FPGA е атаката „предположи и определи“ (guess-and-determine). Тя е интелигентна атака от вида атака на грубата сила. Основната идея на автора ѝ Андерсън е да се отгатне цялото съдържание на регистрите R_1 и R_2 и половината от регистъра R_3. По този начин се определя тактуването на всички три регистри и втората половина на R_3 може да бъде получена от 64 бита на ключовата последователност. В най-лошия случай всеки от общо 2^{52} кандидати трябва да се сравни с ключовата последователност, което изисква голяма изчислителна мощност. Хардуерна реализация на атаката с модификация на идеята на Андерсън е предложена от Келер и Зайц [11]. Те са предложили начин да се изключат значима част от възможните кандидати на много ранен етап от процеса на проверка. Авторите твърдят, че техният подход намалява сложността на атаката до $2^{41}.(3/2)^{11}$ с очаквано компютърно време от 14 тактови цикъла за едно предположение. В резултат на това, в най-лошия случай, сложността е $2^{51.24}$ такта. Атаката е реализирана с Xilinx XC4062 FPGA. Схемата реализира седем паралелно работещи алгоритъма, като работи на честота от 18,65 MHz, което определя време за атаката от около 236 дни.

През 2008 г. Гендрюлис, Новонти и Руп Гендрюлис [7] предлагат модификация на атаката на Келер и Зайц, като изхвърлят грешните възможности за R_3. Сложността на цялата атака е около $2^{54.02}$. Хардуерно устройство COPACOBANA се използва за реализация на атаката. Xilinx Spartan 3-1000 FPGA, с който е реализиран COPACOBANA, съдържа 7 680 елемента, като всеки от съдържа две таблици за търсене LUT (Look-up Table), 2 тригера и други схеми, като бърза логика за пренос или специализирани мултиплексори. Таблиците за търсене могат да се конфигурират като 16 разрядни преместващи регистри. Това позволява всички 15 360 тригера, налични в Spartan 3-1000 FPGA, да реализират 15 360/64 = 240 ядра на шифъра A5/1 без управлението. Реализацията открива вътрешното състояние на A5/1 средно за около 6 часа (в най-лошия случай около 12 часа). Необходими са само 64 последователни бита от ключовия поток без предварително изчисляване на данните.

Атака „компромис между време, памет и данни“. Тази TMDTO атака разкрива вътрешното състояние на A5/1 с определена вероятност в рамките на няколко минути [9]. Използва се отново устройството COPACOBANA, както във фазата на преизчисляване, така и във онлайн фазата на атаката. Избрана е атака с метода **таблици тънка дъга с отличими точки**.

Офлайн фазата на предварително изчисляване може да се изпълни с максимална честота 156 MHz. Всяка стъпка в работата на A5/1 изисква 64 такта. Един FPGA съдържа 234 TMDTO елемента, като всеки се състои от 2 A5/1 ядра при офлайн фазата. Следователно COPACOBANA е в състояние да изпълнява 2^{36} стъпки в секунда при предварителното изчисляване, като всеки един от 120-те FPGA изпълнява около 2^{29} стъпки в секунда. За сравнение, персоналният компютър е в състояние да изпълнява само 2^{22} стъпки в секунда [3].

Атака на шифъра A5/1, реализирана по метода на таблиците тънка дъга, е изпълнена с помощта на хардуера със специално предназначение и ниска цена COPACOBANA, използван по време на двете офлайн и онлайн фази. Понеже онлайн елементът заема по-голяма площ от тази на офлайн, в един FPGA могат да се реализират 160 онлайн елементи. При тази най-висока сложност на дизайна, максималната честотата е 80 MHz. Ако броят на онлайн елементите се намали до 120 е възможно максималната честота да се увеличи до 120 MHz. Вторият вариант е избран от изследователския екип, поради 12,05% по-високата производителност в сравнение с първия. Следователно COPACOBANA е в състояние да изпълнява $2^{34.65}$ стъпки в секунда по време на онлайн фазата. Като се отчете цената на COPACOBANA от 10 000 € и тази на персонален компютър (PC) от 200 € се получава около 330 пъти по-добро съотношение цена-ефективност в полза на COPACOBANA. При

сравняване на изразходваната енергия (COPACOBANA – 600 W, а PC – 150 W) отново се получава около 4 000 пъти по-добро съотношение консумация-ефективност за COPACOBANA. Авторите са избрали набор от параметри, които предоставят разумни срок от 95,4 дена за предварително изчисляване на таблиците и размер на таблиците, което определя капацитет на паметта от 4,85 TB, както и относително малък брой на достъпите до паметта.

Тъй като TMDTO е вероятностна атака, процентът на успех, както и онлайн сложността и броят на достъпите до паметта зависят от броя на прихванатите проби от данни. Ако броят на точките с данни е $D = 64$, тогава успехът е $P = 63\%$ и COPACOBANA изчислява всички резултати за 27,6 секунди, като резултатите се търсят в 10 диска за по-малко от 7 минути. Ако броят на точките с данни се увеличи до $D = 204$, тогава успехът е $P = 96\%$, обаче се увеличава, както времето за изчисляване до 88 секунди, така и времето за търсене в 10 диска до около 22,03 минути.

5. Заключение

С развитието на FPGA технологията, която предоставя много висока изчислителна производителност на сравнително ниска цена, се създаде възможност за използване на преконфигурируемите изчисления за реализация на устройства за криптоанализ на съществуващите шифри. По-ниската цена на еднократната разработка на тези устройства ги направи достъпни и за институции извън правителствените агенции, като научни институти и търговски дружества. Заедно с това, обаче, те стават достъпни и за недоброжелателни групировки. Практическите хардуерни реализации със специализираното устройство COPACOBANA на GSM шифъра A5/1 вече се осъществяват максимално за час, като при по-малка вероятност за успех, те могат да отнемат само минути. Това поставя въпросът за необходимостта от допълнително шифриране при предаване на конфиденциални и чувствителни данни в GSM комуникациите.

Литература

[1]. Babbage S.. A Space/Time Tradeoff in Exhaustive Search Attacks on Stream Ciphers. In European Convention on Security and Detection, May 1995.

[2]. Barkan E., E. Biham, and A. Shamir. Rigorous Bounds on Cryptanalytic Time/Memory Tradeoffs. In Proc. of CRYPTO'06, volume 4117 of LNCS, pages 1-21. Springer, 2006.

[3]. Barkan E., E. Biham, and N. Keller. Instant Ciphertext-only Cryptanalysis of GSM Encrypted Communication (full-version). Technical Report CS-2006-07, Technion, 2006.

[4]. Biryukov A. and A. Shamir. Cryptanalytic time/memory/data tradeoffs for stream ciphers. In Proc. of Asiacrypt'00, volume 1976 of LNCS, pages 1-13. Springer, 2000.

[5]. Briceno M., I. Goldberg, and D. Wagner. A Pedagogical Implementation of the GSM A5/1 and A5/2 \voice privacy" Encryption Algorithms, 1999.

[6]. Denning D. E. R.. Cryptography and Data Security. Addison-Wesley, 1982.

[7]. Gendrullis T. and all. A Real-World Attack Breaking A5/1 within Hours. In Proceedings of the 10th Workshop on Cryptographic Hardware and Embedded Systems (CHES 2008), pages 266-282, Springer Verlag, Washington, D.C., 2008.

[8]. Golic. J. Cryptanalysis of Alleged A5 Stream Cipher. In Proc. of Eurocrypt'97, volume 1233 of LNCS, pages 239-255. Springer-Verlag, 1997.

[9]. Guneysu T., T. Kasper, M. Novotny, C. Paar and A. Rupp. Cryptanalysis with COPACOBANA. In IEEE Transactions on Computers, 2008, vol. 57, No. 11, ps 1498-1513.

[10]. Hellman M. E.. A Cryptanalytic Time-Memory Trade-off. IEEE Transactions on Information Theory, 26:401-406, 1980.

[11]. Keller J. and B. Seitz. A Hardware-Based Attack on the A5/1 Stream Cipher, 2001.
[12]. Kumar S. and all. Breaking Ciphers with COPACOBANA - A Cost-Optimized Parallel Code Breaker. In Proceedings of CHES'06, volume 4249 of LNCS, ps 101-118. Springer-Verlag, 2006.
[13]. Oechslin P.. Making a Faster Cryptanalytic Time-Memory Trade-Off. In Proc. of CRYPTO'03, volume 2729 of LNCS, pages 617-630. Springer, 2003.

За контакти:
ас. инж. Антония Т. Ташева
катедра „Компютърни системи”
Технически университет – София
E-mail: atasheva@tu-sofia.bg

доц. д-р инж. Жанета Н. Ташева
катедра „Комуникационни и информационни системи“
НВУ „В. Левски“, Факултет „А, ПВО и КИС“ - Шумен
E-mail: zh.tasheva@mail.bg

СЕКЦИЯ 2
СОФТУЕРНИ И ИНТЕРНЕТ ТЕХНОЛОГИИ

Юбилейна научна конференция с международно участие
45 години катедра "Компютърни науки и технологии"
30 години специалност "Компютърни системи и технологии"
27-28 септември, 2013 г.
Варна, България

Anniversary Scientific International Conference
45 Years Computer Sciences and Engineering Department
30 Years Computer Systems and Technologies Speciality
27-28 September, 2013
Varna, Bulgaria

SECTION 2
SOFTWARE AND INTERNET TECHNOLOGIES

PERFORMANCE EFFICIENCY OF NATURAL VS. SURROGATE KEYS IN A DATA WAREHOUSE ENVIRONMENT

Ivan Pavlov

Abstract: Modern data warehouses use almost ubiquitously surrogate keys as a cornerstone of the physical data architecture. We analyze experimentally the performance of surrogate vs. natural keys using the industry standard TPC-DS database and typical reporting SQL queries with varied execution characteristics. Our results partially confirm the generally accepted principle that surrogate keys provide enhanced query performance.
Keywords: databases, data warehousing, surrogate key

Бързодействие на естествените и сурогатните ключове в складове от данни

Иван Павлов

Резюме: Модерните складове от данни почти винаги използват сурогатни ключове като основна част от физическата архитектура на данните. Ние сравняваме експериментално бързодействието на сурогатните и естествените ключове, използвайки TPC-DS базата данни и SQL заявки с различни характеристики на изпъление. Нашите резултати частично потвърждават широко приетия принцип, че сурогатните ключове повишават бързодействието на заявките.
Ключови думи: бази данни, складове от данни, сурогатни ключове

1. Introduction

Unlike OLTP systems which often rely predominantly on natural keys, modern data warehouse architecture is based on surrogate keys. The concept of immutable surrogate key has been defined back in 1976 by Hall, Owlett and Todd (see [4]) as a solution to the following perceived problems:

- Situations where updates to the natural keys are needed but without losing the identity of the updated object.
- Distinguishing different states of objects that happen to coexist in the system and are represented by the same natural key.
- Creating an object instance before it exists in the real-world.

In everyday database parlance those advantages translate to benefits for schema management, historization, and the protection of referential integrity in a data warehouse. There is also the widespread belief that integer surrogate keys offer better performance than character-based natural keys. In fact an underlying principle of data warehouse design is that joins between fact tables and dimension tables are always done on surrogate keys. From a more general relation theory perspective the advantages and disadvantages of both key types are discussed in [3].

Limited research have been directed at the experimental comparison of both key architectures. In [1] the authors use an artificial relational database schema to analyze average query speed in the cases of Insert, Update, natural key modification, and Select SQL statements. They find that insert and update statements work better in a natural key architecture while modification of natural key works much better in a surrogate key architecture. Their results in the case of data retrieval operations are inconclusive and depend on the query type. The queries used however are too simple to be of much practical value.

In [5] a simple relational schema is used with a single join statement. The database table sizes are apparently too small and the reported performance differences between the surrogate and the natural key setup do not look significant.

In this article we focus exclusively on the performance of data retrieval operations and use a modified TPC-DS database (see [2]) as the underlying database schema physically realized on an Oracle 12c database. TPC-DS is a decision support industry benchmark that models a real-world data warehouse for the retailing business. It has two main advantages over contrived relational schemas for the problem at hand:

- It contains a large volume of data and provides scripts which can populate the database to any needed size.
- It gives answers to real-world business questions and allows queries of various complexities and execution profiles.

2. Database setup

A limited subset of the tables and fields used in the following sections is presented on Figure 1. We have modified the original database schema by adding to some of the fact tables natural key fields along with the surrogate key fields. Previous research does not mention specific control over the indexing and optimizer statistics of the natural and surrogate key fields which undermines the value of the comparison of data retrieval speeds. We explicitly make sure that the natural key fields have exactly the same indexing and optimizer statistics characteristics as their surrogate counterparts. This allows us in most cases to compare execution speed for similar execution plans.

As much as possible an effort has been made to control the effects of the various caching mechanisms in Oracle on the performance. Multiple query runs and time averaging has been used to control the other factors that may influence runtimes. Besides, the queries has been executed in such a way that if caching and buffering are not eliminated their effects will favor the queries that use natural keys.

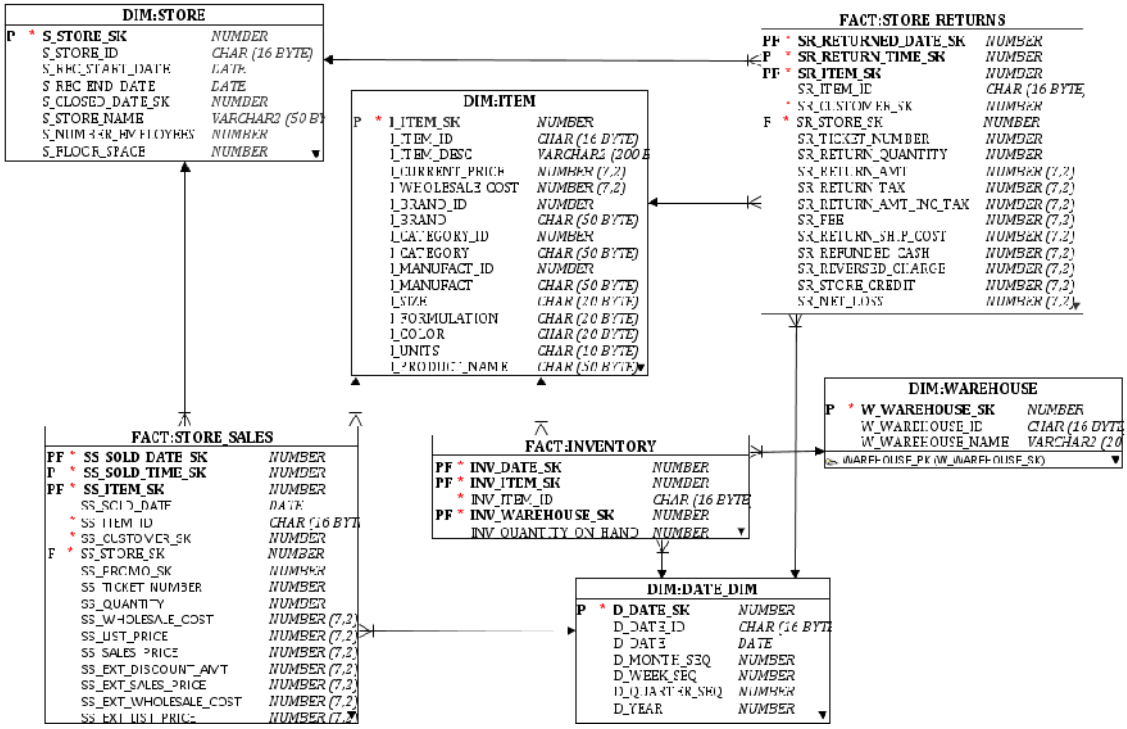


Figure 1. A subset of the TPC-DS database

We focus on several kinds of queries, borrowing some from [2] and others from the official set of SQL queries provided by TPC-DS. The kinds of queries examined are:

- IO intensive queries
- CPU intensive queries
- Reporting queries
- Queries joining multiple fact tables

The data retrieval queries were run on Oracle 12c on a virtualized Oracle Linux R6, with a dual core 1.6 GHz Intel processor and 2 GB of RAM.

3. Experiments

The queries have been run on database tables which have the following sizes:

Table 1. TPC-DS table sizes used

Table Name	Row Count
Inventory	6,000,000
Store_sales	1,500,000
Catalog_sales	60,000
Catalog_returns	76,000
Item	10,000
Date_dim	75,000

The queries chosen all had runtimes of at least 2 seconds. We have restricted ourselves to testing the performance of hash joins (HJ) and nested loop joins (NL). Merge joins on key columns are seldom relevant in a data warehousing environment and thus we did not include them in the query set.

The table below summarizes the results of dozens of independent runs. The "total joins" column indicates the number of joins in the query. In the first variant all those joins were made on surrogate keys. In the second variant each query is rewritten and a number of the joins were made on natural keys - this number is indicated by the "NK joins" column.

Table 2. Query and results description

Query Type	Total joins	NK Joins	Join strategy	Result
CPU intensive	4	4	HJ	SK better: 25-75%
IO intensive	3	3	HJ	SK better: 20-100%
Reporting	3	2	HJ	SK better: 10-30%
Reporting	3	1	HJ	Inconclusive
Simple aggregation	2	1	HJ	Inconclusive
Complex reporting	5	1	HJ	Inconclusive
Simple query	3	2	HJ	SK better: 5-10%
Complex reporting	10	4	HJ	Inconclusive
Star-join	15	2+1	HJ+NL	Inconclusive
CPU intensive	11	0+3	HJ+NL	Inconclusive
Simple join	1	1	NL	SK better: 70-100%

4. Discussion

The presented results point to the following conclusions. For queries in which almost all or all joins are made on natural key columns and the join strategy used by the optimizer is hash join, there is a significant performance degradation. This is even more true when the joins are made using the nested loop algorithm. In this case the joins on natural keys are sometimes twice as slow. In other situations, where only part of the joins are made on natural keys, there is no evidence of any performance degradation. This is probably because the input of the operations involving natural keys is relatively small. The conclusion which can be drawn is that using a surrogate key architecture in data warehousing offers performance benefits which are sometimes significant.

In order to be able to execute the TPC-DS queries without significant alteration we had to remove the historical data from the dimension tables. It can be argued that one of the main functions of surrogate keys is to support exactly this type of data historization in the dimension tables. It may be interesting to repeat the experiments allowing for history records in the dimension tables and rewriting the "natural key"-version of the queries to account for this.

Another unexplored area is the performance of natural vs. surrogate keys in analytical database systems like Teradata or SAP Hana. The architecture of such database systems will probably eliminate all effects the choice of key architecture might have on query performance.

Literature

[1]. Aleksic, S., M. Celikovic, S. Link, I. Lukovic, P. Mogin. "Faceoff: surrogate vs. natural keys". In Advances in Databases and Information Systems, Springer Berlin Heidelberg, 2011, pp. 543-546

[2]. Poess, M., R. Nambiar, D. Walrath. "Why you should run TPC-DS: a workload analysis". In Proceedings of the 33rd international conference on Very large data bases, 2007, pp. 1138-1149.

[3]. Wieringa R., W. De Jonge "Object Identifiers, Keys, and Surrogates: Object Identifiers Revisited", Theory and Practice of Object Systems, 1(2), 1995, pp.101-114.

[4]. Hall P., J.Owlett, S.Todd. "Relations and entities".In G.M.Nijssen, editor, Modelling in Database Management Systems, North-Holland,1976, pp. 201- 220.

[5]. Dragos-Paul, P. O. P. "Natural versus Surrogate Keys. Performance and Usability." Database Systems (2011), pp.55-63

For contacts:
Ivan P. Pavlov
Adastra Corporation
E-mail: ivan.pavlov@gmail.com

TU-Varna FCA Faculty of Computing & Automation

МОДЕЛИРАНЕ НА СЪДЪРЖАТЕЛНО-БАЗИРАНИ СИСТЕМИ, ВЪЗСТАНОВЯВАЩИ ИЗОБРАЖЕНИЯ

Марияна Цв. Стоева, Виолета Т. Божикова

Резюме: Докладът представя опит да се осигури обобщена архитектурна рамка за проектиране на системи, осигуряващи съдържателно-базиран достъп до база данни от изображения, йерархичен модел за изграждане на базата данни с изображения, отчитащи определени техни свойства и формализация на описанието на съдържанието на изображенията, както и мярката за подобие между две изображения. Представените архитектурна рамка предлага избор при проектирането от всички възможни типове компоненти при изграждането на база данни от изображения. Представен е и пример на предлагания от нас йерархичен модел на описание на съдържанието на изображение.
Ключови думи: съдържателно базирани възстановяващи изображения системи, бази данни от изображения, моделиране на данни, моделиране на системи

Content based image retrieval system modeling

Mariana Ts. Stoeva, Violeta T. Bozhikova

Abstract: The need for content-based access to image and video information from media archives has captured the attention of researchers in the recent years. Research efforts have led to the development of methods that provide access to image and video data. Our work tries to provide modelers with a framework and corresponding models for building an image database from a collection of images. We propose a framework and block diagram offering a choice of all possible components in order to build up an image database from a corpus of images and users' requirements. We describe some examples which support understanding of our models.
Keywords: content-based image retrieval systems, image databases, image retrieval, data modeling, system modeling

1. Introduction

Content-Based Image Retrieval (CBIR) systems are in the focus of attention of all visual information systems investigators. New visual query specification methods, new indexes for data setting, and new ways for similarity determination between the saved in the Image Data Bases (IDB) image description data and describing queries [6] are used for image search and retrieval on the base of their contents. The last investigations examine the content based IDB in the context of the object relation model and the Query-by-example philosophy.

The basic directions for the content based IDB development according to [1] are:

- creation of new methods for image description and image property extraction;
- development of efficient data models that allow content based direct organized access;
- elaboration of the image similarity search and retrieval methods;
- formulation of similarity measures that are necessary in the process of images indexing and retrieval; formulation of queries and complex queries processing.

The key directions for each realization are [5]: efficient description and effective extraction; flexibility and extend ability of the capabilities of the content based access; efficiency and utility.

In this paper we are trying, by using the achievements in the area of the CBIR systems, to create a generalized framework model of the CBIR systems that should be valid for the different

types of realizations and applications. Our work tries to provide modelers with a framework and corresponding models for building an image database from a collection of images. We propose a framework and block diagram offering a choice of all possible components in order to build up an image database from a corpus of images and users' requirements. We hope that our work will contribute for the creation of comprehensive environment for modeling and prototyping of CBIR systems and for the development of methodology for image database engineering. Section 2 presents a review of the principle expected features that a framework should include and describes its architecture. An example application is presented in Section 3. The paper finishes with conclusions.

2. Content-based image databases modeling

Image databases designers use three main inputs: an image collection, user's requirements, and an application domain. Each input induces constraints on the databases to be built. Modelers must find a compromise between these constraints.

In order to determine the actual needs of IDB designers, we have studied various projects [2], [3], [6]. Our review of IDB can be summarized as follows:

- Volume of data IDB has to manage huge amounts of data. They use two main strategies (indexing and classification) in order to virtually diminish the amount of data to be researched during the image retrieval.
- An image is described by a combination of syntactical (color histograms, textures, shapes) and semantical (also called meta-data) information and a query can be expressed using a combination of syntactical and semantical image features.
- Granularity image descriptions are generally composed of global and local information.
- Most image databases offer a dual search interface (using either classical queries or query-by-content

The existing systems have been designed by a careful choice of basic mechanisms (indexing or classification) and features of images (syntactical or semantical). Such a choice depends on the database application domain, on images themselves (i.e. on their main characteristics), and on users' requirements.

Our conviction is that an efficient framework should:

- be generic enough in order to cover most of modelers requirements, whatever their image corpus and their users' requirements may be.
- provide a convenient support for both syntactical and semantical information
- provide a convenient support for local and global descriptions.

The creation of our framework of CBIR system steps on the base of the published in [2], [3], [4] models but it differs from them by structure, components, and data models. It includes: built-by-modules subsystems, processing and algorithms types, data models. This framework offers a choice of all possible components in order to build up an image database from a corpus of images and users' requirements.

As a result of the analysis of the existing IDB, of their structures, organization and abilities for access that they give, a generalized architecture of the possible image data processing in the basic processes of saving and retrieval of IDB images is presented. This generalized our architecture-based framework for building an Image Databases from a Collection of Images. It is displayed on Fig.1. and may be used for presentation of the key editions and methods that are proposed in the literature.

In our method's architecture, the basic components are the two mutually connected processes: image saving and inserting in IDB and image retrieval by user's query. The two basic subsystems

„DB Generation Subsystem" and "DB Retrieval Subsystem" correspond to these main processes. In the first subsystem, the images that are added to IDB are processed by features extraction algorithms or by expert. In this way the images are described by syntactical and semantical attributes, which are presented in our data model. The data model includes connected sub-descriptions on different levels. Each one of them includes attributes, a set of components and spatial relations between components. The types of the extracted image features, their describing characteristics and the levels of their extraction (image, object, components) are determining for the abilities of access to the saved data.

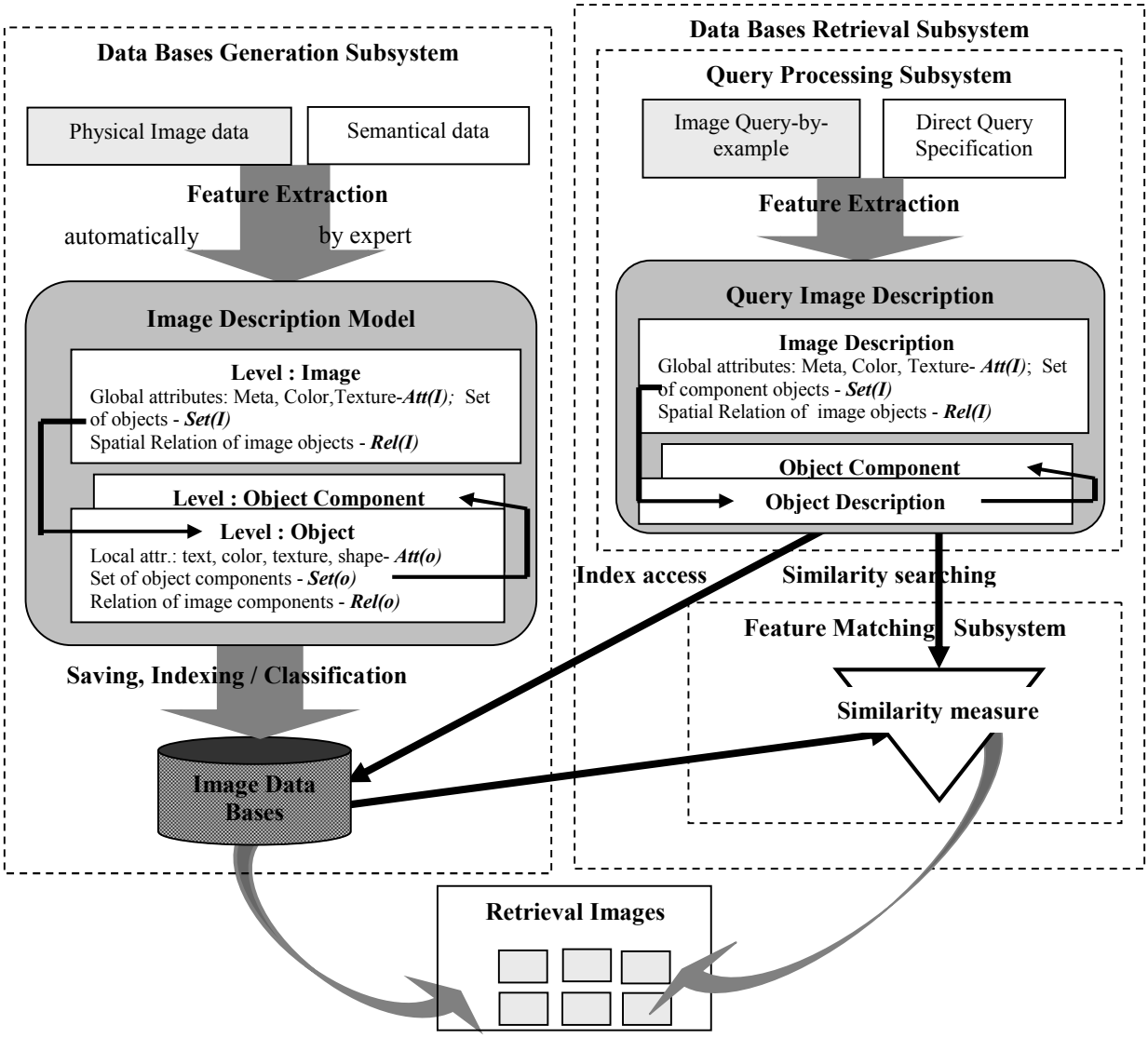


Fig. 1. The framework architecture of CBIR design

Within our model an image, denoted by I is described in terms of simple and of complex objects. Let us denote O_I a set of simple and complex objects. Each object $o \in O_I$ is described by Attributes (semantic and syntactical (colour, texture, shape) denoted by *Att(o)*), Set of object components *Set(o),* and Spatial Relation between Objects *Rel(o)*). We denoted by $Descr_I$ the set of image description: $Descr_I = \{< Att(o), Set(o), Rel(o) >\}_{o \in OI}$

The images inserted in IDB together with their feature description data form a property presentation that is saved in a data structure of IDB. The features in the form of coded vectors can be used as indexes for direct organization of the access to IDB or for data clustering.

According to the level of the used presentation of image contents two approaches for indexing techniques may be determined: indexing on the base of global distribution of the image

characteristics and indexing on the base of the typical peculiarities of local image areas or regions. The small computing complexity and the precision and exactness of the retrieval process are achieved by integration of text annotation with the images as in [6], [5].

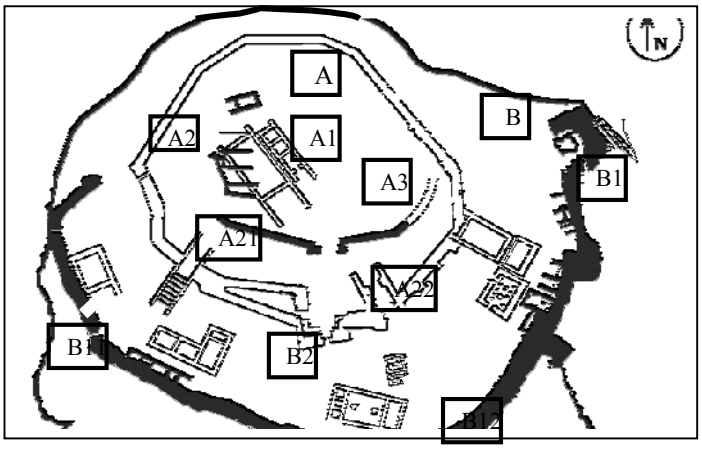


Fig. 2. Image I_{plan} - Plan of ancient town Troy

The „DB Retrieval Subsystem" has to search and discover an answer of user's query. This subsystem is divided in two subsystems: „Query Processing Subsystem" and "Feature Matching Subsystem". This process is determining the system rapidity. The first subsystem processes the query primarily. The query specification may be done by example image, drawn by the user's draft, or exact and clear information from the user about the primary features of his interest. The cognitively based presentation has an important role in query processing on different levels. The query presentation is a result of the same processing for the same properties extraction so as inserting image in IDB. The same algorithms for properties extraction are used also for the query image named "Example" and the result is a presentation that is used for the query index forming. The search of a similar index to those in the IDB is implemented by the "Feature Matching Subsystem". The similarity matching of the sample index with the IDB indexes aims parts of the images to be found that are similar to a given sample or to a defined variant of a given sample. The type and depth of the properties extraction from the inserted in IDB images are determining for the functionality and flexibility of every visual information system. As in most cases the extracted characteristics-indexes are multi-dimensional; the approach of similarity search is perceived. The similarity search uses a similarity measure as a similarity criterion. The measure evaluates the similarity degree between two images and is determining in the process indexing and similarity integration of multi-dimensional index vectors of the image and the query.

In accordance with our data model we can define in a generous aspect the similarity measure that evaluates similarity between query image (Q) description and a saved in IDB image (I) description. The measure sim(Q,I)∈ [0,1] and is given by (1).

$$\mathrm{Sim(Q,I)} = \frac{\left\|O_I \cap O_Q\right\| - \sum_{k}^{\left\|O_I \cap O_Q\right\|} \sqrt{w_1 \sum_{i=1}^{att.number} \left[Att_i(O_{kI}) - Att_i(O_{kQ})\right]^2 + w_2 \sum_{j}^{rel.number} \left[\mathrm{Re}l_j(O_{kI}) - \mathrm{Re}l_j(O_{kQ})\right]^2}}{\left\|O_I \cup O_Q\right\|} \quad (1)$$

where w_1, w_2 are weigh coefficients and $w_1 + w_2 = 1$; $\left\|O_I \cap O_Q\right\|$ is the number of common objects and $\left\|O_I \cap O_Q\right\|$ is the number of all objects in compared images. In a case of absolute identity Sim(Q,I)=1 and is 0 in absence of common objects. Those images from IDB whose presentation is evaluated by the similarity measure as maximal similar are returned as a query result.

As an example for illustration of our frame work architecture and data model we present an image (Fig. 2) denoted by I_{plan} which is a plan of ancient town Troy presented in the archeological literature. The principal of image decomposition is depicted in Fig. 3.

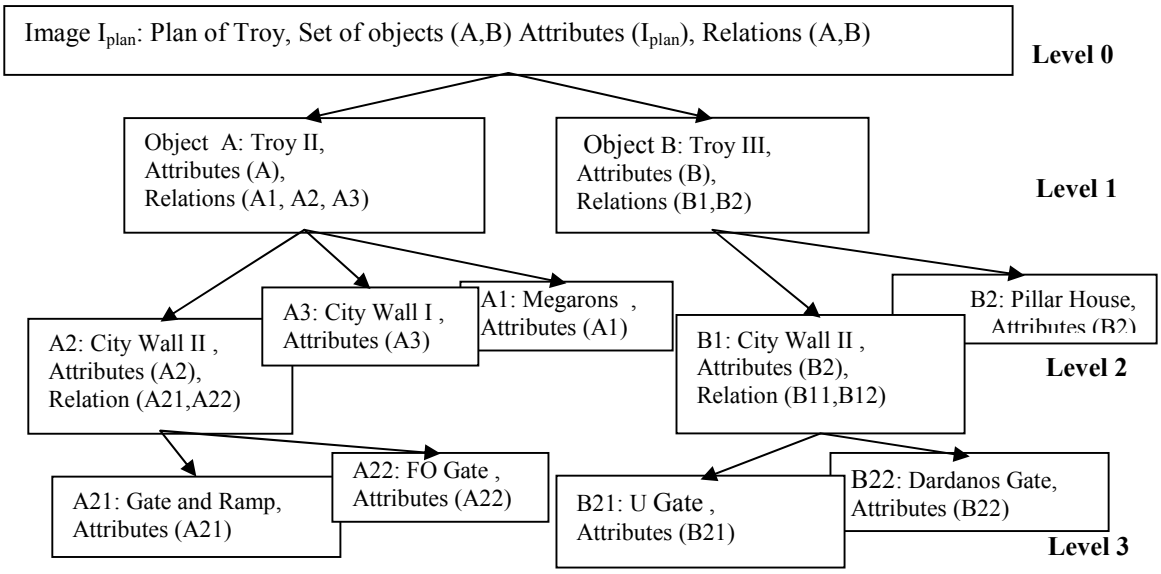


Fig. 3. Hierarchical composition of the image I_{plan}

In the first level there is image description with global attributes. Set of objects includes two complex objects A (Troy II) and B (Troy III). Their descriptions compose the next level 1. By their sides these objects consist also of separate components. In the level 2 are presented components of object A - A1(Megarons), A2 (City Wall II) and A3(City Wall I). The next level 3 contains descriptions of components (A21: Gate and Ramp), A22: FO Gate). Analogically the object B is decomposed to two objects B1(City Wall II) and B2(Pillar House) and B1 contains B11 (U Gate), B12(Dardanos Gate). For the example image I_{plan} we obtain the set of simple objects:

Level 2(I_{plan})={ A1,A21, A22, A3,B11, B12, B2}

Under the control of an expert, a set of all objects can be defined:

OI_{plan} ={I, A, A1,A2,A21, A22, A3,B,B1,B11, B12, B2}

Each object is presented by its description. *$Descr_o$ =< Att(o), Set(o), Rel(o)>*
For example the object A description can be presented by records:

$Descr_A$ =<Attributes "Troy II", area geometry, Sub- objects A1, A2. A3, Spatial relation matrix 3×3)>

3. Conclusions

In this paper, we proposed a framework and a block diagram offering a choice of all possible components for building an image database from a collection of images together with our data model. Image descriptions in our model are based on set of simple and complex objects. Objects can be either syntactical or semantical described in different levels commonly with their spatial relations. With our framework architecture based our and another's experience we try to make the CBIR systems design more understandable. Despite their specificity, we are trying to bring their design to closer to this of the ordinary object-relation systems.

The future development of this work we see in the development of the methodology for image database engineering by using of new methods, approaches and strategies. We already began the creation of a program tool that will poses different feature extraction algorithms, indexing and classification algorithms and combination strategies that are organized in a library.

References

[1]. Al-Khatib, W., Day, Y., F. Ghafoor, A., Berta, P. (1998) Semantic Modeling and Knowledge Representation in Multimedia Databases, IEEE Trans. on Knowledge and Data Ingenering, 11, 1, 64-80

[2]. Antani, S., Kasturi, R., Jain, R. (2002) A surveyon the use of pattern recognition methods for abstraction, indexing and retrieval of images and video, Pergamon, Pattern Recognition, 35, 945–965

[3]. Besson, L., da Costa, A., Lecrcq, E., Terrasse, M. (2003) A CBIR-Framework: Using both Syntactical and Semantical Information for Image Description, Proceedings of the 7th International Database Engineering and Applications Symposium (IDEAS 2003), short paper, IEEE Computer Society, 385-390, Hong Kong

[4]. Besson, L., da Costa, A., Lecrcq, E., Terrasse, M. (2004) Ikosem's Genric Model for Syntactical and Semantical Image description: the core component of a framework for image database engineering", Proceedings of the Tenth International Workshop on Multimedia Information Systems, University of Maryland, USA

[5]. Smith, J.R., () Integrated Spatial and Feature Image Systems: Retrieval, Analisis and Compression, http: //disney.ctr.columbia.edu /jrsthesis /thesissmall. html

[6]. Stanchev, P., (2001) Content-Based Image Retrieval Systems, ComSysTech'2001- Bulgarian Computer Science Conference Sofia, Bulgaria, P11-P15

За контакти:
доц. д-р инж. Марияна Цв. Стоева
катедра КНТ
ТУ-Варна
E-mail: mariana_stoeva@abv.bg

доц. д-р инж. Виолета Т. Божикова
катедра КНТ
ТУ-Варна
E-mail: vbojikova2000@yahoo.com

ОПЕРАЦИОННИТЕ СИСТЕМИ ЗА РЕАЛНО ВРЕМЕ – ТЕКУЩО СЪСТОЯНИЕ

Гриша В. Спасов, Огнян К. Обретенов

Резюме: С напредването на технологиите и увеличаването на изчислителната мощ на процесорите все по-широко приложение намират операционните системи за реално време (RTOS) както в специализирани системи за автоматизация, така и в PC-базирани системи, а в последно време и във вградени микропроцесорни системи. Във все повече устройства, като мобилни телефони, таблети, игри, автоматизация на дома и т.н., се използват системи за реално време, като изискванията към тях стават все по-големи. Настоящата статия има за цел да направи преглед на операционните системи за реално време, като разгледа и сравни тяхната функционалност.
Ключови думи: операционни системи, реално време, вградени системи, разпределение на задачи, приоритети, PC-базирани системи, LINUX, RTAI, RTOS, QNX, WIN CE, VxWorks

Real time operationg systems – state of art

Grisha V. Spasov, Ognyan K. Obretenov

Abstract: With technology development and the increased processors' computing power the operating systems working in real time (RTOS) are more widely used, as well as in specialized automation systems and in PC-based systems, and furthermore, recently in integrated microprocessor systems . In more and more devices like mobile phones, tablets, game consoles, home automation, etc. are used operating systems in real-time and the requirements for them become larger. This article aims to provide an overview for real-time operating systems and to examine and compare their functionality.
Keywords: operation systems, real-time, embedded systems, scheduler, tasks, priority, PC-based systems, LINUX, RTAI, RTOS, QNX, WIN CE, VxWorks

1. Увод

Развитието на технологиите през последните години и напредването на компютърната техника доведе до появата на все по-бързи и по-малки микропроцесорни системи. Те от своя страна направиха възможно все повече разработчици да си поставят и реализират задачи, свързани със системи за реално време.

Ако преди време системите за реално време изглеждаха скъпи и труднодостъпни за широко използване, то сега това не е така. Увеличаване на производителността, вградената памет и броя на ядрата процесори, улесни разработчиците на RTOS да удовлетворят изискванията за системите с реално време с все по-широк набор от процесори [14]. Това доведе използването на RTOS, като ОС, във все повече и повече устройства.

Основната цел на една операционна система е да предоставя възможност на разработчиците на приложения да имат лесен достъп и начин за управление на ресурсите на системата, като осигурят също така и надеждност на работа, модулност, логическа коректност. Добавянето на изискването една ОС да е за реално време води до нова дефиниция, в която освен коректност на резултатите се изисква и тяхната навременност [1].

В зависимост от възможността на системата да реагира/дава резултати по отношение на понятието НАВРЕМЕННОСТ и ефекта от неизпълнението му може да се направи следната класификация: силно критична (hard RTOS), умерено критична (firm RTOS) и слабо критична (soft RTOS) операционна система за реално време [1]. При различните системи

неизпълнението на изискването за време на отговор води съответно до следните резултати и поведение:

- При hard RTOS всяко неизпълнение на дадено изискване води до срив и неработоспособност на системата. Такива операционни системи намират приложение в областта на: бордови системи в автомобилната и авиационната промишленост, управление на реактори, системи за управление на въздушния трафик и др. .
- При firm RTOS, към изискваното време за отговор се допускат ограничени отклонения, но системата като цяло допуска „малки“ пропуски, които не нарушават нормалното функциониране.
- При soft RTOS са допустими нарушения на изискванията за времето на отговор, което води до влошаване на производителността, но не и до срив на системата. Намират приложение в: мултимедиините системи, мобилните комуникации, мрежовите приложения, компютърните игри и д.р.

За да се осигури на приложенията, работещи под операционните системи за реално време, постигане на изискванията за навременност и предсказуемост, трябва да се добави нова по-специална функционалност, както към традиционните, използвани в стандартните ОС компоненти, така и нови специфични за RTOS: управление на паметта, управление на задачите, синхронизация, входно-изходни операции, механизъм за таймери, прекъсвания и събития, комуникация между процеси.

Това е важно както за специализираните RTOS, (като VxWorks), така и за надстройките и разширенията на стандартни ОС до RTOS (RTAI).

Освен тези механизми за RTOS са важни и някои количествени параметри: *латентен период (Latency), неравномерност на отговор (Jitter)* и *най-голямо време за отговор (Worst Case Response Time).*

Целта на настоящата статия е да направи сравнителен обзор на някои от разпространените системи за реално време по отношение на гореизброените механизми и количествените параметри.

Статията е организирана както следва: описание на начини за количествен и функционален анализ на RTOS, кратко описание на RTOS, представяне на сравнителни характеристики, заключение.

2. Параметри и функционалности при RTOS

2.1. Параметрите, даващи количествен оценка

Анализът и сравнението на количествените параметри на RTOS е сложна задача, поради факта, че резултатите са силно обвързани с хардуерните устройства, използвани в тестовете [2].

В литературата се анализират различни количествени характеристики, на които различните автори дават различна важност. Съгласно [3] най-важният количествен параметър за RTOS е „най-голямото време за отговор“ (Worst response time), по който може да се квалифицира системата като hard, firm или soft. Според Tanenbaum [11] съгласно този параметър RTOS са само два вида (hard и soft). Това време е в пряка връзка с общото натоварване на системата. Съгласно [4] най-важен параметър е времето, през което прекъсванията за забранени. Все пак може да се отделят три основни параметъра, около които се обединяват повечето автори:

- Латентен период (Latency) – този параметър се изследва като системата за реално време се разглежда като черна кутия. Измерва се времето от момента на възникване на събитието до момента на генериране на отговор.

• Неравномерност на отговор (Jitter) - това е времето от момента на възникване на събитието до момента на поемане на управлението от съответната обработваща програма. Този параметър е много важен за системи, които въвеждат данни, зависими от времето. Типичен пример са системите с ЦПУ при металорежещите машини, където през определено време се чете позицията, до която са достигнали работните органи на машината, и на база на тези данни се прави корекция на заданието. Неравномерността при тези системи води до влошаване на равномерността на движение [15] [16].

• Най-голямо време за отговор (Worst Case Response Time) – това е параметър, от който зависи как ще класифицираме системата и като цяло ще се използва за оценка дали тази система може да изпълнява определена задача. Оценката на този параметър е сложна задача, защото трябва да се анализират и предвидят всички възможни ситуации, които могат да влияят на времето на отговор. Това са както софтуерни (програмна забрана на прекъсванията), така и хардуерни ситуации като работа с DMA.

2.2. Компоненти и функционалности на RTOS

При сравнението на системите за реално време от особено значение са функционалностите на компонентите, които системата поддържа и осигурява на приложния софтуер.

При сравнението ще се разгледат следните компоненти:

• Управление на задачите (Multitasking) – при системите за реално време се прави ясно разграничение между „планирани“ и „непланирани“ задачи. Всяка планирана задача се характеризира с определен контролен блок и при нея предварително са ясни изискванията за ресурси. Тези задачи се управляват от система за разпределение (Scheduler). Системата за разпределение и задачи, обслужващи прекъсвания или системни извиквания, са „непланирани“ задачи. Тези задачи поемат пълен контрол на системата и могат да се изпълняват непрекъснато, периодично или по събитие. Изискване към управление на задачите и превключването им е да е с възможно минимално време, да поддържа освен стандартните механизми за превключване, базирани на приоритети и на преразпределение, но и специализирани за RTOS като RR (round robin) [11]

• Синхронизация е важен механизъм за всяка операционна система, позволяващ задачите да използват споделено определени ресурси, като памет, буфери, хардуерни устройства. Обикновено се използват семафори и мутекси за синхронизация. Но при RTOS трябва да се подобри този механизъм като се избегне основния проблем за така нареченото обръщане на приоритета. Това е състояние, когато високоприоритетна задача чака ресурс от нископриоритетна. Това при RTOS може да доведе до критични резултати в отговора на дадено събитие. За това при RTOS трябва да се търси използването на допълнителни техники за синхронизация като RIP (*priority inheritance protocol)* [13] или PCP (*priority ceiling protocol*). Тези техники позволяват избягване на проблемите с приоритетите.

• Управление на прекъсванията - тъй като RTOS винаги са свързани с управление на хардуер, механизмът за прекъсвания е от особено голямо значение. Системата трябва да предоставя лесен начин на разработчиците да пишат задачи (tasks) за обслужване на софтуерни и хардуерни прекъсвания [1].

• Управления на броячи и часовници – този механизъм, както и прекъсванията, трябва да е много добре поддържан. Изискват се функции за броячи, таймери и часовници, както и достъп до вградените в процесорите часовници за наносекунди времена.

• Управление на комуникацията между задачи - поддържаните механизми за обмен на данни между задачите както в ОС, така и в RTOS са: обща памет, сокети, опашки от съобщения, именовани канали (pipes) и FIFOs (файл вход, файл изход) [16]. Изискванията при системите с реално време към гореизброените механизми са наличие на неблокираща и

асинхронна комуникация и операции с определено време. Напоследък тази комуникация се разширява от една RTOS към разпределени RTOS.

- Управление на паметта – това е механизъм, който е неотменна част от стандартните ОС, но е проблематичен механизъм в RTOS, защото един процес може да бъде блокиран от липса на памет или времето за отговор да стане много голямо при една операция за прехвърляне на страница от диск в памет. Системите за малките процесори не поддържат този механизъм, но повечето процесори имат вградени в себе си MMU (memory management unit), така че някои от RTOS предлагат и този механизъм [1].
- Поддържани процесорни платформи - тази характеристика на ОС дава възможност на разработчиците да намалят времето и разходите за разработка на системи за реално време. Повечето съвременни RTOS поддържат системи на база на процесори ARM, MIPS и x86.

3. Общ преглед на избрани (популярни) RTOS

3.1 Microsoft Windows CE Embedded 6.0

Последната версия вече е под името Windows Embedded Compact 7, тук е разгледана предходната версия Win CE Embedded 6.0. Тя поддържа системи, базирани на архитектури ARM, MIPS, SH4 и x86.

WinCe показва много добра стабилност при всякакви натоварвания. Платформата предоставя няколко практически инструмента за проверка на системата дали отговаря на изискванията за реално време. Такива средства са: Kernel Tracker, ILTiming и OSBench. Те дават възможност и за дебъгване на задачи. Интересна функционалност е също превенцията на системата от претоварване. При опит да се запуснат множество задачи, системата следи натоварването и дава съобщение за претоварване и предотвратява пускане на нови задачи [9].

3.2 QNX Neutrino

QNX Neutrino е една от най-разпространените RTOS, която предлага „определяемо време за отговор“. Тази система се базира на архитектурата “микроядра”, което позволява много голяма надеждност и превенция. Проблеми и сривове на задачи, които работят в ядрото, не води до срив на цялото ядро и система, както при системите с обикновени ядра [6].

3.3 Real Time Application Interface (RTAI)

Тази система, която се явява надстройка/модификация на стандартно Linux ядро, получи широко разпространение при малки приложения, в тестови и изследователски задачи, като и в медицината. Тя е система с отворен код (open source), което й дава много добро съотношение цена/качество. Съгласно много изследователи тази система може да реши над 90% от задачите за реално време в съчетание на това, че като е Linux базирана, то може да се използват и всички останали стандартни приложения под Linux [7].

RTAI модифицира стандартното ядро като създава области, изолира стандартната система в област, създава друга област за реалното време и прави тази област с най-висок приоритет. Също така създава микропланировчик, който разпределя времето между реалновремевите задачи и останалите. Системата за планиране позволява работата както на реалновремевите задачи, така и на останалите задачи от ОС.

RTAI като RTOS има добрите резултати, получени при тестовете, но като основен недостатък може да се даде липсата на гаранции и сертификати за качествата и надеждността на системата.

3.4 Wind River VxWorks

VxWorks е една от най-използваните RTOS във вградените системи, също и в космонавтиката [5]. Тя е сертифицирана от няколко агенции, че отговаря на стандартите за системи за реално време, надеждност и критични приложения [8].

Ядрото на системата позволява задачите да бъдат групирани по функции и да са изолирани в защитени структури. Тази система е с най-добрите количествени параметри.

Като недостатък може да се каже, че е сравнително по-трудна за инсталация и конфигурация, съгласно тестовете и анализите на dedicated-systems.info [12].

4. Резултати от сравнение на параметри и характеристики на RTOS

4.1 Количествени характеристики

Количествените характеристики са базирани на изследванията на Rafael Aroca [2]. Измерванията са направени на една и съща платформа PII 400mhz, 256 Мб RAM. В таблица 1 са дадени критичните стойности от измерванията на трите времеви параметъра на четири популярни RTOS. От направените измервания най-добри показатели за системи, работещи в силно критично (hard) реално време, показват RTAI и VxWorks.

Таблица 1 – Измерени времена

	WinCE	Qnx	RTAI	VxWorks
Време за отговор	20.00 мкс	20.00 мкс	5.00 мкс	3.85 мкс
Латентност	99.00 мкс	35.20 мкс	11.40 мкс	13.40 мкс
Неравномерност	88.80 мкс	32.00 мкс	7.01 мкс	10.40 мкс

4.2 Функционални характеристики

Функционалните характеристики са базирани на изследванията, направени от Ramesh Yerraballi [10] и документацията на всяка от операционните системи, взети от техните фирмени сайтове, както следва: Windows CE [9], за RTAI [7], за QnX [6], за VxWorks [8].

4.2.1 Планиране и многозадачност

Разгледани са какви нива приоритети се поддържат и дали са позволени нишки. Планирането на задачите използва механизмите FCFS (First Come First Serve) и RR (Round Robin). FCFS не е подходящ за RTOS, защото един процес може да заеме ресурса за много време. RR е подобен на FCFS, но премахва описания недостатък на FCFS, като независимо от необходимото време за задачата, RR дава само по един квант време на всички процеси, като ги обхожда циклично. Всички разглежадни системи използват планиране на задачите с фиксиран приоритет, RR и една нишка. Само RTAI и VxWorks има по-добри параметри по отношение на броя на приоритетите в сравнение с останалите системи.

4.2.2 Управление на прекъсвания и събития

Управлението на прекъсването (Interrupt Service Routines –ISRs) оказва голямо значение за производителността на системата. По-добрите системи предлагат както възможност за управление на хардуерните прекъсвания, така и софтуерни (POSIX) прекъсвания/събития. Най-добра функционалност предлагат Qnx и VxWorks, поради това, че те предлагат и възможност за вложени прекъсвания.

4.2.3 Синхронизации

Механизмите на синхронизация позволяват на задачите да споделят общи ресурси. Всички системи, освен RTAI, използват стандартните механизми, като семафори, мутекси и протоколи RIP или PCP/HLP. Тук най-слаби параметри има RTAI. При тази RTOS се използва FIFO за синхронизация.

4.2.4 Междупроцесна комуникация

Основните механизми за междупроцесорна комуникация са: обща памет, канали, опашки, съобщения, FIFOs. С разширението от една към разпределена система се появяват и сокети и RPC, които се използват като за локална комуникация и така и за обмен в разпределени системи. WinCE предлага само „обща памет‘, RTAI предлага и FIFOs, като най-много механизми за комуникация предлагат системите Qnx (Съобщения, Канали, Опашки, Сокети) и VxWorks (Съобщения, Канали, Опашки, Сокети , Обща памет, RPC).

4.2.5 Управление на памет

От разглежданите системи само RTAI не предлага тези механизъм (Вуртуална памет и динамично заделяне на памет), но въпреки това общата препоръка към разработчиците е да се избягва използването на тези механизми, поради възможността при грешка в логиката на потребителската задача да се компрометира времето за отговор.

4.2.6 Поддържани платформи

Поддръжката на повече хардуерни платформи дава повече възможности на разработчиците да бъдат по гъвкави в избора на хардуер и да мултиплицират разработките си. Всички RTOS подържат множество процесорни платформи (x86, ARM, SH4, MIPS) и броят им непрекъснато се увеличава.

5. Заключение и бъдеща работа

Безспорно изборът на RTOS е сложна задача, която зависи от цена, функционалности, хардуер, време на разработка, сложност на задачата, време за отговор и т.н. Всяка една от RTOS се развива с времето, добавят се нови функционалности, поддържани платформи, а също така се появяват нови RTOS.

Анализирайки резултатите, може да се каже, че всички от разглежданите RTOS разполагат с много богат арсенал от механизми и функции за реализиране на задачи с реално времеви функции. С най-добри параметри безспорен лидер е VxWorks, но не всяка фирма може да си я позволи, а също така и не всяка задача има достатъчен бюджет за тази RTOS.

RTAI се оказва една разумна алтернатива, особено за научно-развойни цели.

Бъдеща задача е да се анализират по-задълбочено реализациите на Linux базираните RTOS. Да се подбере подходяща Linux базирана RTOS и да се реализират и тестват приложения за реално-времева задача, свързана с обработка на данни от ЦПУ. Също така и да се анализират възможности за намаляване на влиянието на латентността и неравномерността при въвеждането, обработката и извеждане на резултат от ЦПУ.

Литература

[1]. Laplante, P. A. (2004). Real-time systems design and analysis : an engineer’s handbook , Third Edition, IEEE Press 2004, ISBN: 0-471-22855-9.

[2]. Rafael V., Aroca G. C., “A Real Time Operating Systems (RTOS) Comparison. In: Workshop deSistemas Operacionais (WSO)”, Congresso da Sociedade Brasileira de Computação (CSBC), 2009. Bento Gonçalves, Brazil 22-23 July 2009.
[3]. Sohal, V. (2001). How to really measure real-time. *Embedded System Conference.*
[4]. Labrosse J., “MicroC/OS-II - The Real Time Kernel”, CMP Books 2002, ISBN-10: 1578201039.
[5]. Cedeno W., “An overview of real-time operating systems”, Journal of the Association for Laboratory Automation, 12 (2007) , pp. 40–45, ISSN: 1535-5535
[6]. QnX . (n.d.). Retrieved from http://www.qnx.com/.
[7]. RTIA . (n.d.). Retrieved from https://www.rtai.org/.
[8]. VxWorks. (n.d.). Retrieved from http://www.windriver.com/products/vxworks/.
[9]. Windows Embeded. (n.d.). Retrieved from http://www.microsoft.com/windowsembedded/en-us/developers.aspx.
[10]. Yerraballi R. (n.d.), “Real-Time Operating Systems: An Ongoing Review”, Proceedings of 21st IEEE Real-Time Symposium, WIP sectione Systems, Orlando Florida, 2000.
[11]. Tanenbaum A., “Modern Operating Systems” (3rd Edition), Prentice-Hall 2007, ISBN-10: 0136006639,
[12]. COMPARISON BETWEEN QNX RTOS V6.1, VXWORKS AE 1.1 AND WINDOWS CE, Dedicated Systems Experts, 2002. (http://www.dedicatedsystems.com)
[13]. Mall R., “Real-Time Systems, Theory and Practis”, Pearson Education 2008, ISBN 9788131700693.
[14]. Stellwag Philippe, Schroeder-Preikschat Wolfgang, “Challenges in Real-Time Synchronization”, Proceedings of the 3rd USENIX Workshop on Hot Topics in Parallelism (HotPar '11) Berkeley 2011, CA, USA 26.-27.05.2011, стр. 1-6
[15]. Erwiński, Krystian; Paprocki, Marcin ; Grzesiak, Lech; Karwowski, Kazimierz ; Wawrzak, Andrzej, “Application of Ethernet Powerlink for communication in a Linux RTAI open CNC control system”, IEEE Transactions on Industrial Informatics, Volume:60 , Issue: 2, 2012, pp. 628 – 636, ISSN: 0278-0046.
[16]. Обретенов Огнян, Георгиев Веселин, “CANOPEN ПРОТОКОЛ ЗА ВРЪЗКА НА PC-БАЗИРАНО ЦПУ С ЦИФРОВИ СЕРВО УСТРОЙСТВА”, Journal of the Technical University Sofia, branch Plovdiv “Fundamental Sciences and Applications”, Vol. 14, 2009, pp. 373-378, ISSN 1310 – 8271.

За контакти:
проф. д-р Гриша В. Спасов
катедра „Компютърни Системи”
Технически Университет София-филиал Пловдив
e-mail: gvs@tu-plovdiv.bg

гл. асистент Огнян К. Обретенов
катедра „Компютърни Системи”
Технически университет - София - филиал Пловдив
e-mail: oko@unisoft-eng.com

ЗАЩИТА И ПРЕМАХВАНЕ НА ВИРУСИ ОТ USB УСТРОЙСТВО

Петя В. Белева – Димитрова

Резюме: В настоящия доклад се разглеждат начини да се открият вируси на преносими USB устройства, способни да заразят компютъра още с отварянето им (след като устройствата бъдат свързани към компютъра). Също така са дадени практически описания как да се процедира при откриване на вирус на такова устройство – как да се премахне вируса и как да се защити устройството от последващо заразяване с този вид вируси. Операционната система, използвана за описанията, е Windows 8.
Ключови думи: защита, вирус, autorun.inf

Protection and removing viruses from USB drive

Petya V. Beleva – Dimitrova

Abstract: In this paper, some ways how to discover viruses in portable USB devices are considered (these viruses can infect the computer when they be connected to). Also, practical descriptions how to remove the virus and to protect the device are given. The operational system used for the descriptions is Windows 8.
Keywords: protection, virus, autorun.inf

1. Увод

Когато използваме интернет, вирусите винаги са причина за безпокойство. Съществуват много начини, чрез които компютърната система може да се зарази със злонамерен софтуер. Един от тях е да се използва вече заразено флаш устройство. Известни вируси като *„Ravmon“, „svchost.exe“, „Orkut is banned“, „Autorun.inf“, „Recycler“, „autorun.vbs“, „kavo.exe“, „Heap41a“, „newfolder.exe“, „newfolder.com“, „ppt.exe“* и др. се разпространяват чрез USB устройства, но това може да се избегне, ако USB устройството е защитено срещу запис (*write protected*). Това не е задължително, ако на компютъра е инсталирана добра антивирусна защита, но не всички антивирусни програми могат да открият тези вируси, а ако го направят, невинаги са способни да ги изтрият.

2. Предпазване на компютри и USB устройства

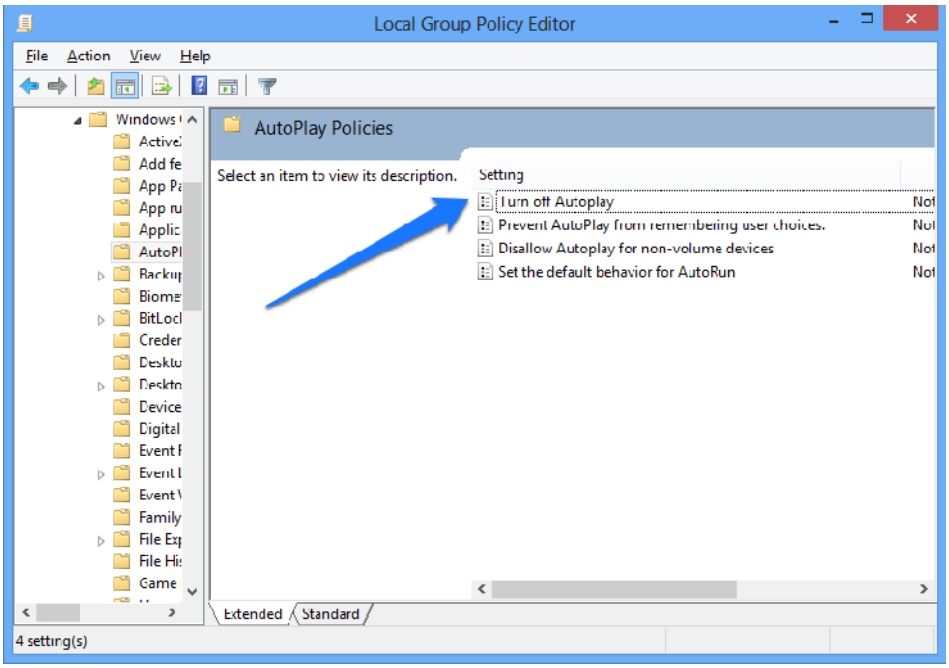
Ето и няколко начина, чрез които може да се предпази информацията на компютъра и USB устройствата.

- ***Забрана на AUTORUN за WINDOWS 8***

Забраната на автоматичното стартиране на USB устройство предпазва и от автоматичното стартиране на вирусите, което може да зарази системата. За да стане това, се прави следното:

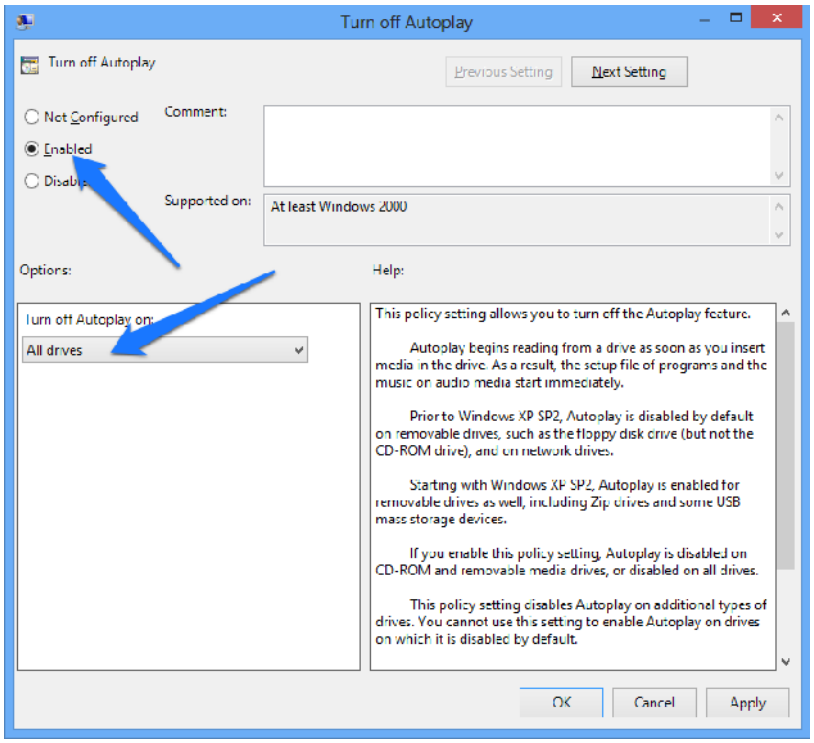
1. За да се появи кутията Run, се използва клавишна комбинация **Windows Logo+R,** след което се изписва **gpedit.msc** и се натиска OK.
2. Отваря се диалогов прозорец **Group Policy Editor**, а от там се избира: *Computer Configuration >> Administrative Templates >> Windows Components >> AutoPlay Policies*

3. В десния панел се натиска двукратно върху **Turn off Autoplay** (както е показано на фиг. 1)



Фиг. 1. Group Policy Editor

4. Избира се **Enabled** и **All drives** и **Apply**:

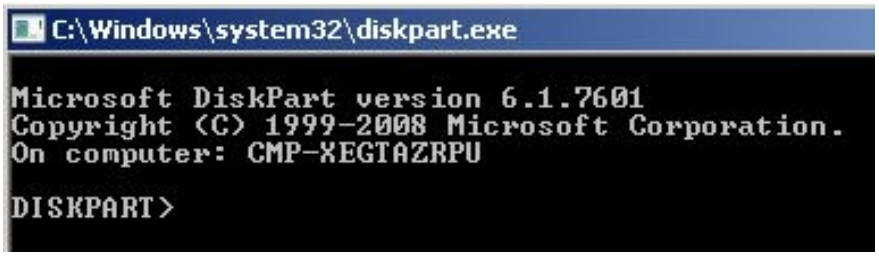


Фиг. 2. Group Policy Editor

- ***Защита на USB устройство срещу запис***

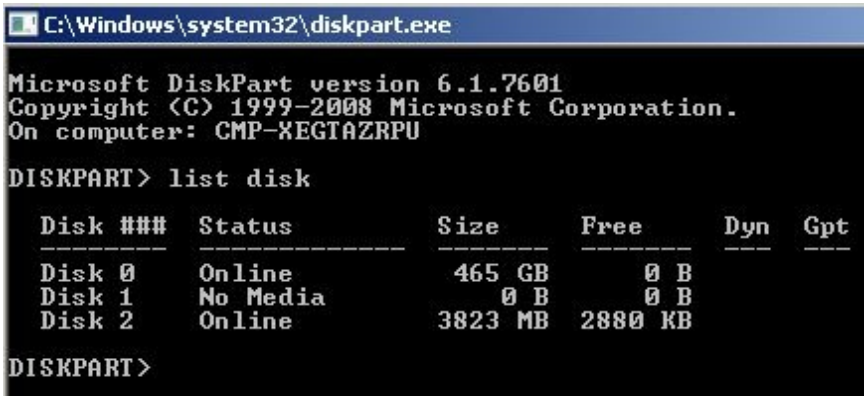
За да се защити USB устройство срещу запис, е необходимо в полето RUN да се изпише ***diskpart*** (това е вградена програма в Windows, която създава, модифицира, изтрива или

променя дялове и защитава от запис), след което се отваря прозорецът, показан на фигура 3 [1][3].



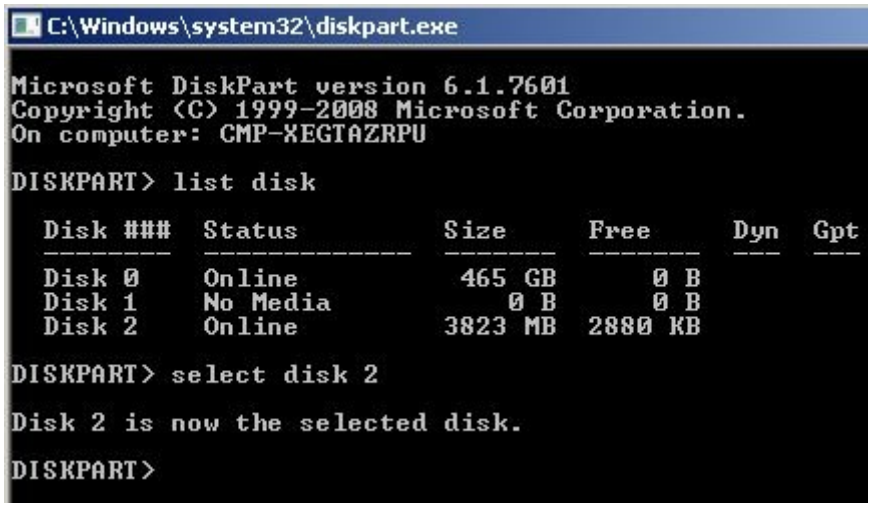
Фиг. 3. Прозорец на Diskpart

Въвежда се командата, която ще покаже списък, съдържащ устройствата на компютъра (твърди дискове и мобилни USB устройства) – ***list disk.*** Показва се подобен на фиг. 4 прозорец.



Фиг. 4

В случая мобилното устройство (флаш памет) е Disk 2. Трябва много добре да се запомни кой е номерът на диска на флаш паметта и да се въведе командата ***select disk x*** (на мястото на x се записва номерът на USB устройството – в случая – select disk 2), след което се показва подобен на фиг. 5 прозорец.



Фиг. 5

След като USB устройството е избрано, за да се защити от запис (write protected) се въвежда команда *attributes disk set readonly*, след което се показва подобен на фиг. 6 прозорец.

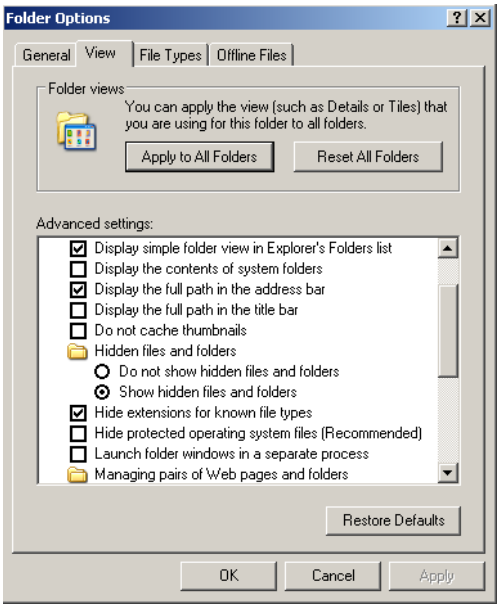
Фиг. 6

След изпълнението на тези стъпки, на флаш паметта няма да може да се запише нищо и ще излиза съобщение, че е само за четене. Ако след това се налага запис върху нея, се използва команда ***attributes disk clear readonly*.**

- ***Изтриване на вируса Autorun.inf***

Когато се постави USB устройство, се показва съобщение за автоматично стартиране. Ако на флаш устройството има вируси, това ще зарази компютърната система. Затова винаги трябва да се избира Cancel и първо да се проверява устройството за наличието на вируси. Един от начините за това е : [2][3]

1. Проверява се коя е буквата на USB устройството (в случая е G:);
2. От меню **Start** се избира команда **Run** (за по-кратко може да се използва клавишната комбинация **Win+R**) и се въвежда **cmd**, за да се отвори командният промпт на Windows (Windows command prompt);
3. За да се отвори директорията на USB устройството, се изпълнява командата **cd** \. Показва се, че текущата директория е **C:\>,** след което се изпълнява команда **G:**. За да се види съдържанието на директорията и да се покажат атрибутите на отделните файлове, се изпълнява командата **attrib**. Ако устройството е заразено, ще се видят файлове като ***autorun.inf, autorun.vbs, kavo.exe, vbs, Heap41a, newfolder.exe, com, ppt.exe*** и т.н. Атрибутите на тези файлове обикновено са: само за четене, архивни, системни и скрити. Следващата стъпка ще е да се премахнат. Това става с командата:
4. **G:\> attrib –s –h –r \autorun.inf**; Файлът ***autorun.inf*** обикновено е с **shr** атрибути (**super, hidden, readonly**), а това го прави труден за изтриване. За целта тези атрибути се премахват;
5. Изпълнява се **G:\>attrib**, за да се провери успешно ли са премахнати атрибутите;
6. Следва командата **G:\> dell autorun.inf;**
7. Отново **G:\> attrib**, за се види, че вирусът вече не е в списъка;
8. За да се провери със сигурност дали вирусът е изтрит, е хубаво да се излезе от Windows command prompt и да се извади безопасно USB устройството, след което пак се поставя и се проверява отново със стъпка 3.
9. Друг начин, чрез който може да се открие ***autorun.inf*** е, след като се постави USB устройството, да се направи следното :
10. След натискане с десен бутон на **My Computer** се избира опция **Explore;**
11. Отваряне на устройството от **Windows Explore**;
12. За да се види ***autorun.inf*** е необходимо да няма скрити папки. За целта се избират отметките (както е показано на фиг. 7) във **Folder Options**:



Фиг. 7. Folder Options

- **Изключване на *autorun.inf* за постоянно**

Това ще предотврати бъдеща поява на ***autorun*** червея. Следният текст се копира в notepad и се запазва с разширение ***.reg***., след което се рестартира компютърът. [1][2]

```
REGEDIT5
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\IniFileMapping\Autorun.inf]
@=”@SYS:DoesNotExist”
```

3. Сравнителен анализ

В таблица 1 са представени резултати от направено проучване, относно възможностите на няколко антивирусни програми за откриване на различни видове *autorun* вируси. Версиите на тестваните програми (както и вирусните дефиниции) са актуализирани към 24.08.2013 г. Когато антивирусната програма не засече или само открие вирусите, проверката и почистването им е извършено и тествано с всички изброени методи – премахване посредством графичната среда на операционната система и чрез използването на командния режим.

Таблица 1: Резултати относно възможностите на някои антивирусни програми

	Avast	**F-secure**	**AVG**	**Avira AntiVir Personal**	**Microsoft Security Essentials**
Autorun.inf	открива	открива	не открива	не открива	открива и изчиства
Newfolder.exe	открива	открива	не открива	не открива	открива и изчиства
Recycler	открива и изчиства	открива и изчиства	не открива	не открива	открива и изчиства
Svchost.exe	открива	открива	не открива	не открива	открива
ppt.exe	открива	открива	не открива	не открива	открива и изчиства

4. Изводи и заключения

Направените проучвания сочат, че от анализа могат да се направят следните изводи:

1. Различните антивирусни програми имат различна ефективност за откриване и изчистване на *autorun* вируси;
2. Предвид факта, че на един компютър не може да се инсталира повече от една антивирусна програма (получава се конфликт между тях), очевидно няма как да се постигне цялостна ефективност само с антивирусен софтуер;
3. Налага се използването на метод за почистване, подобен на описаните в доклада. В зависимост от предпочитанията на съответния потребител, може да се избира работа изцяло в графична среда или безпроблемно да се справи с командния режим на Windows.

Предвид факта, че средностатистическият потребител най-често не може да си позволи използването на антивирусен софтуер като Kaspersky, Norton и други (скъпо платени са и натоварват системата), то не може да се очаква 100% сигурност от страна на антивирусния софтуер. Поради тази причина е добре потребителите да познават поне един начин за откриване и премахване на вируси от преносими USB устройства, както и да имат познания как да се предпазят от тях в бъдеще.

Литература

[1]. Anderson, Br.; Anderson, B., Seven Deadliest USB Attacks, Syngress, 2000, p. 256
[2]. Collins, B., How to Create a USB Thumb Drive Computer - Secure method to perform virus scans and online banking, Kindle Edition, pp. 1-15
[3]. Take Care Avoiding and Removing USB Viruses, IBM bulletine, 2012

За контакти:
докторант Петя В. Белева-Димитрова
катедра „Информационни технологии”
ВТУ „Св. Св. Кирил и Методий“
E-mail: peti.bell@gmail.com

Bulgaria Communications Chapter

AN APPROACH TO CREATING A VIRTUAL 3D MODEL OF A MEDIEVAL TOWN

Stanislav D. Kostadinov, Tzvetomir I. Vassilev

Abstract: This paper presents an approach for 3D reproduction of a medieval town. A procedure for creating a virtual model of a destroyed medieval town is proposed. Technical solutions of API for visualization of 3D models are compared. A few virtual museums are reviewed and new architectural solution for fast and light-weight application for visualization is made. Flexible and easy to learn modelling tool for creation of virtual models is proposed.
Key words: Virtual Museum, Cultural Heritage, VRML, X3D, WebGL, JOGL.

Подход за създаване на виртуален 3D модел на средновековен град

Станислав Д. Костадинов, Цветомир И. Василев

Резюме: Тази статия представя един подход за тримерна репродукция на средноведковен град. Предложена е методика за създаване на виртуален модел на разрушен средновековен град. Сравнени са различни API за визуализация на 3D модели. Разгледани са няколко виртуални музеи и е представено ново архитектурно решение за създаване на бързо и малко приложение за визуализация. Предложен е гъвкав и лесен за употреба инструмент за създаване на тримерни виртуални модели.
Ключови думи: Виртуален музей, Културно наследство, VRML, X3D, WebGL, JOGL.

1. Introduction

In the recent years many scientists in the field of Computer Graphics collaborate with historians to build virtual models of historical buildings and monuments, which were destroyed during the years. Two major projects were started in the last 2 years in Bulgaria that reveals a new look of old historical sites. The first one, started in 2011 is a virtual tour of the ancient Philippopolis which is one of the previous names of Plovdiv, Bulgaria. The other one is the virtual museum of Sofia, Bulgaria. It is part of the candidacy of Sofia for the European cultural capital in 2019. A year ago, team from Great Britain started a project of photographing and cartographing Cherven medieval town. Some high scale images from a weather balloon from different views were made.

The work, described in this paper, is a part of a joint project of computer scientists and researchers at the regional history museum in Ruse, Bulgaria, for creating a virtual 3D model of the medieval town of Cherven, situated near Ruse. During the years the town was destroyed, but multiple excavations were performed and archaeologists have built a relatively detailed plan of city and some buildings.

The rest of the paper is organized as follows. The next section reviews previous work on virtual reconstruction. Section 3 studies the existing software for 3D modelling and existing format and APIs for implementing 3D visualization on the Web. Section 4 describes in details the proposed approach. The last section concludes the paper and gives ideas for future work.

2. Previous work on virtual reconstruction

A few teams have attempted to create virtual museums in the early 90's. First two of them are the museums in New York and Paris. They contain high resolution images and do not match the phrase 'virtual museum' as it is used nowadays. Most of the images do not have any accompanying

comments and they are just sorted image sets. In the latest 90s, new site has been created by the Canadian Museum of Civilization [1]. It has been a new phase of Internet museums that include rich text pages associated with the images of the city of New France, Canada.

A next level of the collaboration of technologies and historical artefacts is the natural historical museum of Los Angeles. The team from LA has created a site with huge set of artefacts with intuitive search engine that provides quick access to each museum object. On-line shop and event exhibition plan have been integrated, too. A similar type of site is one of the biggest collections of on-line historical, scientifically and political objects that are situated in the so called ibiblio. It is a digital library and archive with more than 2500 collection hosted on 25 servers.

One of the first virtual museums that contain 3D tour is the Smithsonian museum of national history [2]. The navigation in each museum hall is not very intuitive and the traffic between the client browser and the server is flooded with JPEG images used to refresh the scenes. Newer and more sophisticated way of representing historical buildings or museum halls is the Google art project [3]. The open source project provides high quality image viewer and building navigation through museum halls. Navigation is done by multiple AJAX requests to server for images of each view. This kind of virtual museums does not allow simulation of human walk through the historical monument or real museum hall. It just makes approximations of user's manipulations and provides limited set of views.

One of the pioneers of virtual reconstructions of monuments is Yiorgos Chrysanthou with his reproduction of the old city of Nicosia [4][5].

Fig. 1. Reconstruction of Nicosia by Yiorgos Chrysanthou

In 2012, a team from USA led by Kuzminsky and Gardiner worked on ancient human fossils and made a 3D model of them [6]. That was a new step forward of virtual museums and progress of the quality of the virtual museum representations. Benefits of that museum conservation were that people all over the world have the ability to have access to ancient skeletons in a three dimensional view and it became easier to ordinary people to imagine what artefacts look like in a real scale. Other 3D models of museum artefacts were made by the teams of Fabio Bruno and Stefano Bruno [7] and Fernand Cohen and Zexi Liu [8].

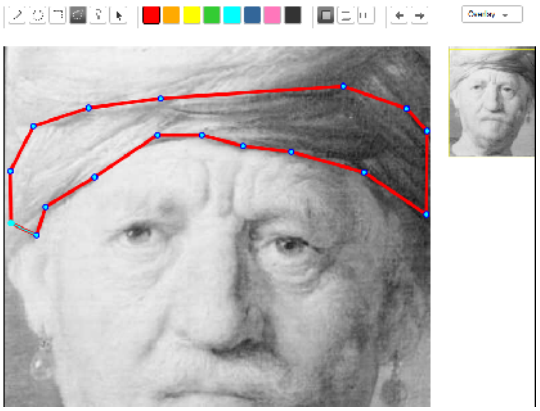


Fig. 2. Semantic annotation in Research space project

2012 was the year in which semantic technologies stepped in the virtual museums. A team from Sirma Group developed application called ResearchSpace [9]. It is a project of the British Museum that was built on the basis of RDF (Resource Description Framework) database management system called OWLIM. Semantic technology enables machines as well as people to understand, share and reason with them at execution time. With semantic technologies, adding, changing and implementing new relationships or interconnecting programs in a different way can be just as simple as changing the external model that these programs share. Next challenge for that team is implementation of ConservationSpace, initiated by National Gallery of Art in Washington, USA.

3. Study of 3D modeling software and API for 3D on the Web

3.1. 3D Modelling software

One of the most powerful design suites were developed in the last few years by Autodesk Company. That manufacturer provides software for building architectural, mechanical, electrical, structural and plumbing systems. They provide CAD systems but the suites that best fit to the needs of creation a virtual model of medieval town are Autodesk 3DStudio MAX and AutoCAD Civil 3D. They are extremely sophisticated and the learning curve is not steep.

3DStudio MAX enables artists and designers to focus more energy on creative, rather than technical challenges. It's more oriented to flexible creation of dynamic animated object rather than in static ones like construction materials. That suite provides too many features, which makes it tough to learn and it makes the whole 3D model too "expensive" of time and human resources. The product allows exporting in gbXML or the so called Green Building XML which is very close to X3D language family. An advantage of 3DS MAX is the big community and the great variety of file formats. Some of the disadvantages are:

- the number of vertices and polygons per mesh is limited to 65536;
- accurate vertex normal vectors cannot be stored in the .3ds file (3DS MAX file extension);
- directional light sources are not supported.

AutoCAD Civil 3D is a part of the Autodesk Infrastructure Design Suite and it provides solutions mostly used by engineers and designers to create buildings and infrastructure. One of the advantages of Civil 3D are the surveying tools which can work with more than 4000 real-world coordinate systems and generation of 3D models for GPS machine control. The suite can create point clouds that are part of surfaces. That will be useful for steep terrains where huge meshes compose ground details. Civil 3D uses tin surfaces that contain triangles only and that makes it hard for maintaining in comparison with rectangular meshes but it is excellent for rough terrain areas because it uses optimal number of points.

Another popular software tool is Rhinoceros. It is commercial NURBS-based 3D modelling software and it is mainly used by architects, marine designers, jewellery designers and automotive designers. The tool provides great variety of formats for saving of the resources. They can be XML-based, JSON-based or special formats like these used by Autodesk tools. That makes Rhinoceros flexible for use and it can export all the data to 3DS MAX format whenever it is necessary. Its advantages are:

- provides mesh modelling tool (other mesh tools can be added with different plug-ins);
- easy to learn in few days for general purpose and it is intuitive in its use;
- simplified and optimal for generation of meshes and basic 3D primitives.

All of the above mentioned solutions provide student licenses and they can be used for the 3D Cherven town model without paying any license taxes. They should provide a tool set for creation of huge meshes and view with algorithms for smoothing them. The algorithms for smoothing should be similar to the one from the museum viewer and its algorithm should be fast [10].

3.2. Formats for 3D on the Web

One of the pioneer formats for description of 2D and 3D scenes is VRML. It is an abbreviation of Virtual Reality Modelling Language. This language provides primitives (parallelepipeds, spheres, cones, cylinders, set of points, set of lines, and set of planes) and its own flexible format of transforming them by rotation, translation and scaling. Complex elements are presented by set of points, lines or polygons. They could be visualized in a few coordinate systems and the connection between the systems could be presented with transformation data the same way like the primitives. VRML provides a way for definition of scripts that can be used for animating the elements from the scene. VRML was mostly used a decade ago and because of its transformation flexibility it was used for animation of animals and architectural constructions [11].

Google developed another JSON-based format called O3D. It was created for interactive 3D graphics and originally it was build in as chrome plug-in. Latest implementation of O3D is a JS library implemented on the top of WebGL. In fact, O3D visualization and interaction with 3D object is integrated with the settings of the objects. That makes the meta data hard for reading and maintaining.

Another format that has a fast-growing community is WebGL. It is similar to O3D because it keeps the source code of rendering the scene and the scene objects in one place. It is partially supported by some of the newest version of the browsers (Firefox, Safari, Chrome) but others like Internet Explorer do not support it at all. A paper of Diego Cantor-Rivera and Terry Peters describes how to use WebGL in medical imaging [12].

Nowadays, XML based formats are the most preferred formats for transferring data through Internet. There are a lot of XML editors and they have high level of compression which is important for decreasing the traffic between server and client. An XML successor that is similar to VRML is X3D. In fact, X3D extends VRML and provides some new primitives like Humanoid animation, NURBS and GeoVRML. X3D allows fast access to its tags (elements) if it is used in combination with DOM or XPath, etc. X3D was used as format for representation of historical sites like Stonehenge [13].

Some of the most innovative geographic information systems use standard called CityGML. It is an XML successor and the abbreviation comes from City Geography Markup Language. Gerhard Groger and Lutz Plumer used it to focus on the semantic model of the 3D models, its structures, taxonomies and aggregations [14]. Since 14-th of March 2012 Open Geospatial Consortium adopted CityGML as an official standard. After that date CityGML became a standard for transferring geographical data through web services.

There is no adopted 3D format for data transfer between a server and a client [15]. VRML and other JSON successors were used in the last decade of the 20-th century but after that XML-based formats became more popular because they are user-friendly, they can be edited easier and a lot of XML editors were made to support developers and even ordinary users.

3.3. API for 3D visualisation on the Web

There are a lot of viewers for X3D or VRML files. Most of them are outdated and not maintained by their developers. Such tools are OpenVRML and FreeWRL. They need installation plus additions settings. They are not compatible with all browsers under all operating systems and that restricts the client.

Google O3D is one of the latest tools used to visualize 3D object on the web. In the beginning it was a plug-in to Chrome but later it was organized as a JavaScript library implemented on the top of WebGL. It could be run on Internet Explorer, Firefox, Chrome and Safari. However, your browser should be compatible with WebGL, it works with a limited set of graphics cards and it has no implementation for mobile devices.

Java applet is good solution of both desktop and web-based applications. There are some fine applications that are based on Java, or example, the applet that renders 3D scene by using the JOGL API. It is an extension of OpenGL used by Java developers. There are some applications with similar architecture. One of them is combined with Xj3D but it is quite slow and complicated for additional use.

Current existing solutions for visualization of 3D object are too big, too slow or have too much hardware restrictions. A useful solution might be a JOGL application. It should be fast and it should generate low traffic between server and client. It will be open source, cross-platform and cross-browser.

4. Proposed approach

The authors of this paper propose the following procedure for creating a virtual 3D model of a destroyed medieval town:

1. Studying existing plans of the town and buildings.
2. Taking aerial photographs of the ground and comparing with the existing plans.
3. Studying the terrain in Google Earth and creating a new accurate plan of the town based on existing maps, air photographs and Google Earth maps.
4. Selection of an appropriate 3D modelling software.
5. Collecting textures for the old buildings, created by artist and designers.
6. Creating the virtual model using the selected software.
7. Developing an application for the visualization. It can be used for both online and standalone version of the system. The customer will be able to navigate through the streets and look inside some of the buildings.

The most flexible 3D modeling software for generation of virtual 3D model of a medieval town is Rhinoceros. The application provides tool for mesh modeling and a set for manipulation which facilitates construction of the complex shapes. The tool provides functionality for export in file formats compatible to other well-known modelling tools which makes it easy to replace it later if necessary. This tool is more oriented to modelling rather than in rendering 3D scenes which fits to the need of the reconstruction. Another useful functionality is creating meshes from point cloud data. This technique speeds up generation of objects part of the 3D model and it is well-known from other reconstructions. Last but not least, this tool exports data in XML format that is simplified and human-readable format and easy to be parsed from any client-side application.

XML successors fit best to this application because they have high compression ratio. X3D is a good solution because it is standardized and other applications can use it too.

The tool for visualization of the virtual museum will be implemented with Java technologies because they are cross-browser and cross-platform. Java applet will be used for the presentation layer and JOGL will be used for rendering 3D objects. SAX parser will be used for parsing of the model file.

5. Conclusions

This paper presented an approach for creation and visualization of a virtual model of a medieval town. Most of the tools and sites do not provide realistic view of the presented historical monuments. Some fast and light-weight solutions were presented in order to implement new application that will satisfy user's environment and needs.

As a conclusion, such application for visualization of virtual models of towns is one step ahead in each direction: loading speed, realism, interaction, cross-platform, cross-browser, model compatibility with other applications and simplicity of model generation.

References

[1]. Gordon B., Archaeological Survey of Canada Paper, Canadian Museum of Civilization Mercury Series, 1 (vol. 28)
[2]. Hoyo-Melendeza J., M. Mecklenburg, An evaluation of daylight distribution as an initial preventive conservation measure at two Smithsonian Institution Museums, Journal of Cultural Heritage, 4 (vol. 14), 2011, pp 54-64
[3]. Proctor N., The Google Art Project: A New Generation of Museums on the Web, Curator: The Museum Journal, 1 (vol. 54), 2011, pp 215–221
[4]. Lerner A., Y. Chrysanthou, D. Cohen-Or, Breaking the Walls: Scene Partitioning and Portal Creation, Proceedings of the 11th Pacific Conference on Computer Graphics and Applications, 1 (vol. 1), 2003, pp 303-309
[5]. Charalambous P., Y. Chrysanthou, A. Charalambous, Reconstruction of everyday life in 19th century Nicosia, Proceedings of the 4th international conference on Progress in Cultural Heritage Preservation, 2012, pp 568-577
[6]. Kuzminsky S., M. Gardiner, Three-dimensional laser scanning: potential uses for museum conservation and scientific research, Journal of Archaeological Science, 8 (vol. 39), 2012, pp 2744-2751
[7]. Bruno F., S. Bruno, From 3D reconstruction to virtual reality: a complete methodology for digital archaeological exhibition, Journal of Cultural Heritage, 1 (vol. 11), 2010, pp 42-49
[8]. Cohen F., Z. Liu, Virtual reconstruction of archaeological vessels using expert priors and intrinsic differential geometry information, Computers and Graphics, 1 (vol. 37), 2013, pp 41-53
[9]. Alexiev V., J. Parvanova, S. Kostadinov, RDF Data and Image Annotations in ResearchSpace, submitted for publication in ACM Symposium on Document Engineering, 2013
[10]. Vassilev T., S. Kostadinov, Planar grid generation for cloth simulation, Proceedings of the University of Ruse Conference, 1 (vol. 1), 2012, pp 53-58
[11]. Cao Z., Z. Hu, Design Virtual Reality Scene Roam for Tour Animations Base on VRML and Java, Physics Procedia, 1 (vol. 25), 2012, pp 693-699
[12]. Cantor-Rivera D., T. Peters, Pervasive medical imaging applications - current challenges and possible alternatives, International Mobile Health Workshop, 2010
[13]. Hetherington R., B. Farrimond, S. Presland, Information rich temporal virtual models using X3D, Computers and Graphics, 2 (vol. 30), 2006, pp 287-298
[14]. Groger G., L. Plumer, CityGML – Interoperable semantic 3D city models, ISPRS Journal of Photogrammetry and Remote Sensing, 1 (vol. 71), 2012, pp 12-33
[15]. Vassilev T., S. Kostadinov, An Approach to 3D on the Web using Java OpenGL, Proceedings of the University of Ruse Conference, 1 (vol. 1), 2009, pp 55-60

For contacts:
MSc Stanislav Dimchev Kostadinov
Department of Informatics
University of Ruse
E-mail: sdkostadinov@uni-ruse.bg

Dr. Tzvetomir Ivanov Vassilev
Department of Informatics
University of Ruse
E-mail: TVassilev@uni-ruse.bg

ТАКСОНОМИЯ НА ДАННИ ГРАДИВНИ БЛОКОВЕ

Васил Б. Милев

Анотация: Разработката на архитектура на предприятието е процес на моделиране и документиране на всички аспекти на организацията. Разработена е таксономия на данни градивните блокове, която се базира на три основни класификационни признака: абстракция, обхват и зрялост. Тя може да послужи при специфициране и съхранение на информация за данни градивните блокове. Разработката може да се прилага като ръководен материал за ИТ отдели, които започват да реализират проекти в областта на синхронизацията на бизнеса и информационните технологии.
Ключови думи: архитектура на предприятия, архитектура на данните, моделиране на данни, таксономия, EIM, 3D Model, eTOM.

Taxonomy for data building block

Vassil B. Milev

Summary: The Enterprise Architecture is the process of modeling and documentation of all aspects of the organization. The taxonomy of "data building block" is developed, which is based on three main classification criteria: abstraction, scope and maturity. It can be used for specifying and storing information for data building blocks. This work can be used as guidance material for IT departments, which will begin to implement projects in the field of synchronization of business and information technology.
Keywords: enterprise architecture, data architecture, data modeling, taxonomy, EIM, 3D Model , eTOM.

1. Въведение

Разработката на архитектура на предприятието [14,19] е процес на моделиране и документиране на всички аспекти на организацията, за да се гарантира, че услуги, процеси, приложения, информация, данни, технологии, места, хора, събития и срокове са съобразени с целите и задачите. Архитектурата е фундаментална организация на системи, реализирана чрез нейните компоненти, връзките помежду им, и с обкръжаващата среда, както и принципите, определящи нейното проектиране и развитие [13]. За да се осигури синхрон между развитието на бизнеса и информационните технологии (ИТ) [11], е необходимо създаване на таксономии [4], удовлетворяващи гъвкавото им развитие.

Обикновено при структуриране на данните се използват стотици или дори хиляди имена на архитектурните елементи от данни. По-голямата част от имената са омоними, синоними и псевдоними, създадени в отсъствието на три фактора:

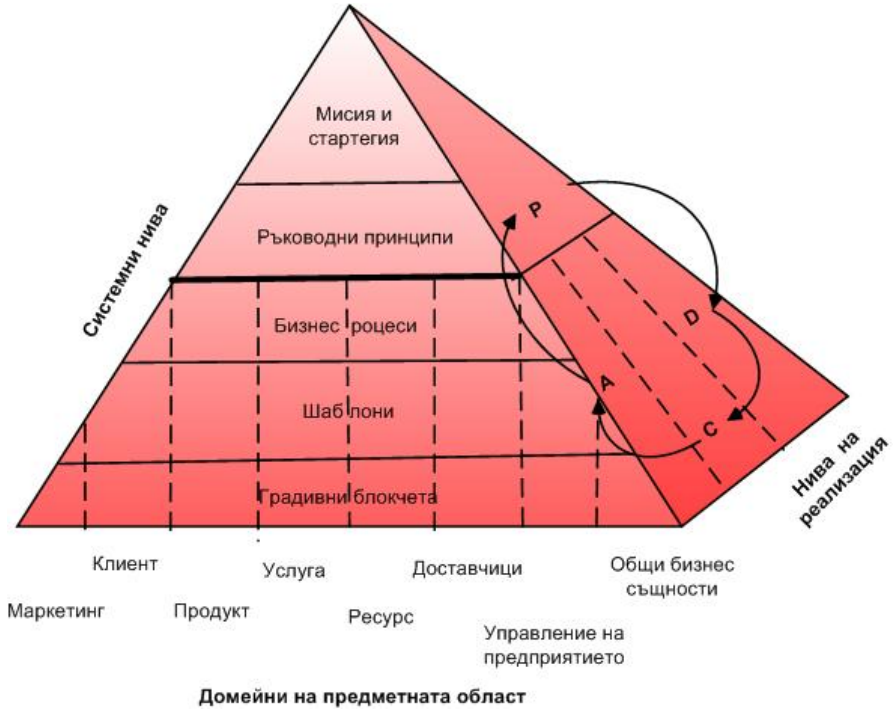
- стандарти за именуване;
- гледна точка на корпоративната архитектура на данни;
- планиране на създаване и използване на данните.

Цел на настоящия доклад е да представи система и средства за решаване на този проблем, т.е. чрез използване на малко на брой типови архитектурни елементи, наречени "данни градивни блокове" (Data Building Blocks, DBB), които са систематизирани, описани и визуализирани, да се оптимизира дейността по моделиране на данни.

Тази цел се реализира чрез разработка на таксономия [4,17] на DBB, която класифицира по определени признаци основните архитектурни компоненти, използвани при моделиране на данните.

2. Място и роля на данни градивните блокове и тяхното описание

Мястото и ролята на DBB се изяснява чрез 3D МОДЕЛ на архитектура на данните [16] и разширения информационен модел (Extended Information Model, EIM) [7], представени на фиг. 1 и фиг. 2. Моделите описват конкретни архитектурни решения в областта на моделиране на данни [6,8], които се явяват типови и могат многократно да бъдат използвани в различни приложения.

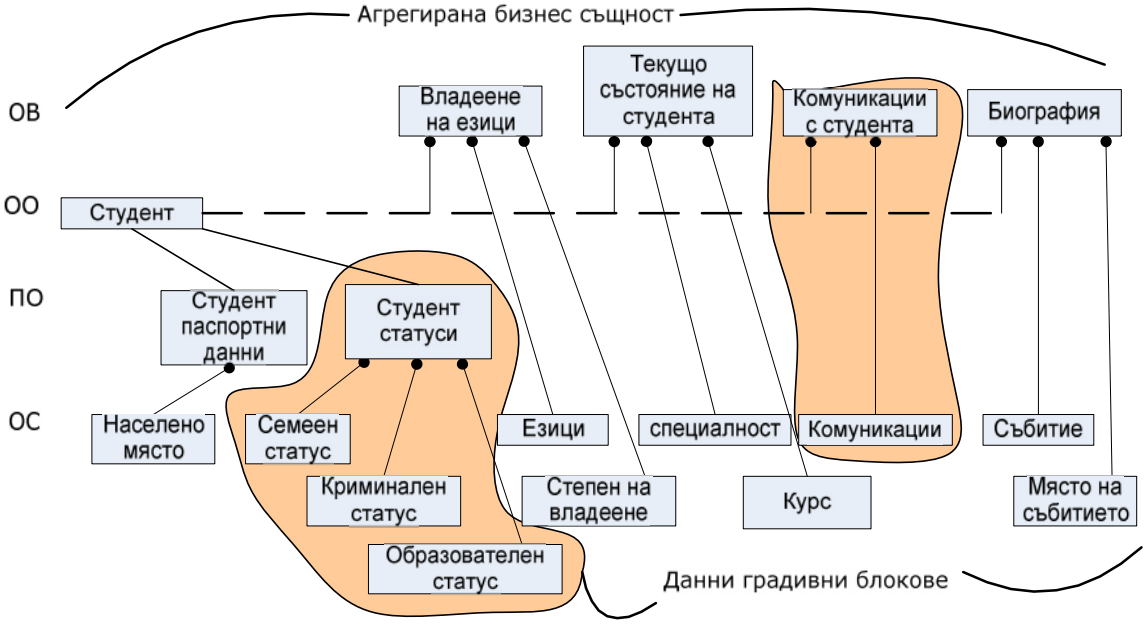


Фиг. 1. 3D модел на архитектура на данните [16].

Фиг. 2. Разширен информационен модел [7] .

Представляват функционално обособени един или група обекти, които реализират конкретна функция в рамките на домейн от конкретна предметна област.

Примери за градивни блокове [9, 10, 15] са представени на фиг. 3: Комуникации със студента, Данни за студента, Статуси на студента в Агрегирана Бизнес Същност (Aggregated Business Entity, ABE), Студент в предметна област - Образование.



Фиг. 3. Агрегирана бизнес същност - Студент.

От изключително значение за разработка на качествени модели на данни е детайлното описание на DBB, което включва:

- извършва се текстово описание на същността;
- изброяват се ABE, в които градивният блок участва;
- изброяват се същностите, които участват в градивния блок;
- описват се правила на използване и реализация.

Пример: Комуникации с персона

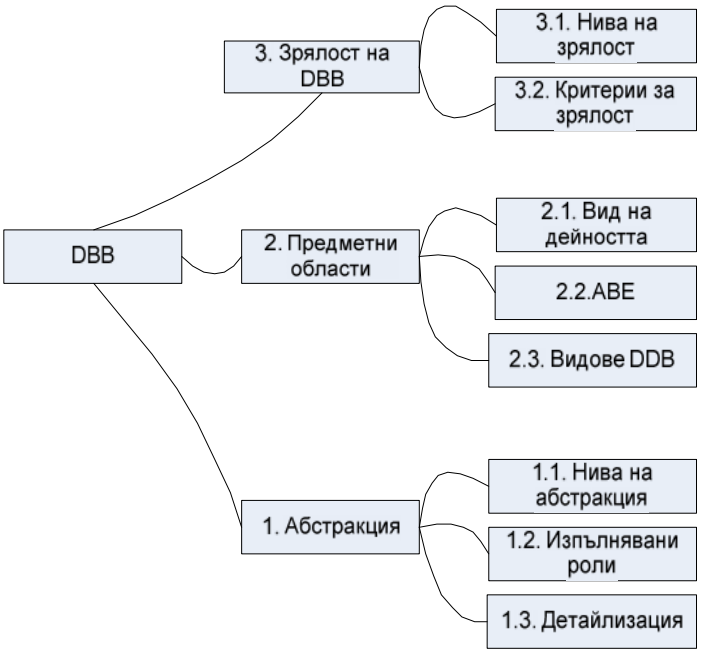
Таблица 1

Название	Комуникации с персона
Описание	Подробно описание на градивния блок
АБС	Студент, преподавател, служител
Свързани същности	Персона, комуникации, комуникации с персоната
Правила	

3. Таксономия на данни градивните блокове

Таксономията (от гръцки τάξις-строй, порядък и νόμος-закон) е наука за класификация, включваща правила, теория, методи и приложението им [4, 17]. Представлява колекция от контролирани условия и лексика, организирани в йерархична структура.

Бизнес таксономиите или така наречените класификации, вече се радват на доверие, като липсващо звено на проекти за управление на данни. Под таксономия в доклада ще се разбира набор от избрани термини, използвани за извличане на онлайн съдържание. Разработената примерна таксономия (фиг. 4) ще се използва при специфициране и съхранение на информация за данни градивните блокове [9].



Фиг. 4. Таксономия на данни градивните блокове.

Основни компоненти на таксономията се явяват:

1. Абстракция

Указва предназначението на DBB, неговия разработчик и собственик, както и реализиращия го модел. Включва следните нива:

1.1. Нива на абстракция - моделирането на данни се реализира в рамките на три модела - концептуален, логически и физически. Данни градивните елементи въпреки и част от общия модел на данни, подлежат на същите обработки.

1.1.1. Концептуален модел - обикновено се създава в информационния етап стратегия за управление. Съдържа ключовите същности и отношения и представя графично, на високо ниво концепцията за дата градивния блок. Не зависи от използваната за реализация платформа и програмен език. Реализира се чрез използване на метода PER (Process-Entity-Relation) [6, 8].

1.1.2. Логически модел - описва конкретните структурни елементи (таблици), техните атрибути и съответните типове данни. Определя приложимите бизнес правила.

1.1.3. Физически модел - крайното представяне на структурите на базата данни, които са получени от модела. Съдържа подробни спецификации за проектиране на бази данни.

1.2. Изпълнявани роли

1.2.1. Архитект - разработва системната архитектурата на предприятие, включваща архитектурата на данните, приложенията и инфраструктурата [13, 19].

1.2.2. Проектант - проектира приложенията и съпътстващите ги бази данни, като създава логическия модел на данните.

1.2.3. Разработчик - разработва приложенията и физическият модел на базата данни.

1.2.4. Собственик - отговаря за съответни данни градивни блокове - съхранение, актуализации, нови приложения.

1.3. Детайлизация

1.3.1. Предприятие/организация - съвкупност от взаимодействащи елементи, които реализират конкретни бизнес процеси и преследват определени цели [13, 14, 19].

1.3.2. Проект - извършване на съвкупност от разнородни дейности за решаване на сложен нестандартен проблем при наложени ограничения относно време, разходи, качество и специфични изисквания към неговата организация.

1.3.3. Приложение - програма за реализиране на конкретна задача.

1.3.4. Модул на приложение - част или фрагмент от приложението, за реализиране на конкретната задача .

2. Предметни области

Основна задача на информационното моделиране е разработка на формално описание на дадена предметна област (обикновено засягаща едно предприятие), независещо от особеностите на реализация на модела и съответното програмно осигуряване. Предметната област касае сфера на дейност в реалния свят, за която е била поставена задачата да бъде прехвърлена в компютърния свят. Примери: телекомуникации, университети, медицина, търговия. Тя се разделя както следва :

2.1. Вид на дейността - за всяка организация са типични три направления в дейността, които се описват със съответните бизнес процеси. Явяват се основните домейни, които трябва да бъдат моделирани при изграждане на информационната инфраструктура и приложения в една организация. Много добре са развити в концепцията The New Generation Operations Systems and Software (NGOSS) на TeleManagment Forum (TMF) [12] и по специално в модела на процеси The enhanced Telecom Operations Map (eTOM) [1].

На 1, 2 и 3 нива от модела eTOM [1] се извършва детайлизация на описаните по-горе области до подпроцеси (например, диагностика на проблеми или контрол на показателите за производителност на ресурсите):

2.1.1. Стратегия [6,7,8] - включва целия спектър въпроси свързани със стратегията на организацията, развитието на инфраструктурата и управление на жизнения цикъл на продуктите (услугите).

2.1.2. Операционни процеси [7,12] - явяват се най-важните процеси в една организация, които обезпечават представянето и осигуряването на услуги и продукти.

2.1.3. Управление на организацията - обхваща общи въпроси по осигуряване дейността на организацията, такива като управление на кадри, финанси и активи, знания, външни връзки и т.н.

2.2. Агрегирани бизнес същности [7]

Представляват тясно свързани различни обекти, обединени в една група. Тя се състои от DBB, които изясняват и детайлизират основната бизнес същност-обекти, участващи в съответния процес. ABE се явяват основни за предприятието данни, които зависят от сферата, в която то действа. Пример за ABE в областта на образованието са: студент, преподавател, дисциплина, ресурси и т.н.

2.3. Видове градивни блокове

В модела SID (Shared information and data model) [2,12] за всяка ABE са определени следните данни градивни блокове.

2.3.1. Основни данни - описва основните данни на ABE, като определя същността на останалите обекти.

2.3.2. Стратегия и планиране - описва задачи, критерии и срокове за изпълнение на решения.

2.3.3. Спецификация - описва различни статуси на ABE.

2.3.4. Комуникация - отразява данни, касаещи мобилни, стационарни и интернет комуникации.

2.3.5. Конфигурация - описва вътрешната структура на ABE - паспортни данни, фирмени данни, вход, изход и т.н.

2.3.6. Параметри за ефективност - описва параметри за оценка на производителността (ефективността) качеството на ABE.

2.3.7. Средства за тестване - описва средства и инструменти за оценка на ABE.

2.3.8. Проблеми - описва проблеми, които касаят ABE.

2.3.9. Събития във времето - описва събития и тяхната историческа свързаност в контекста на ABE.

В таблица 2 са дадени примери за данни градивни блокове на “Преподавател” и “Студент” за ABE.

Таблица 2

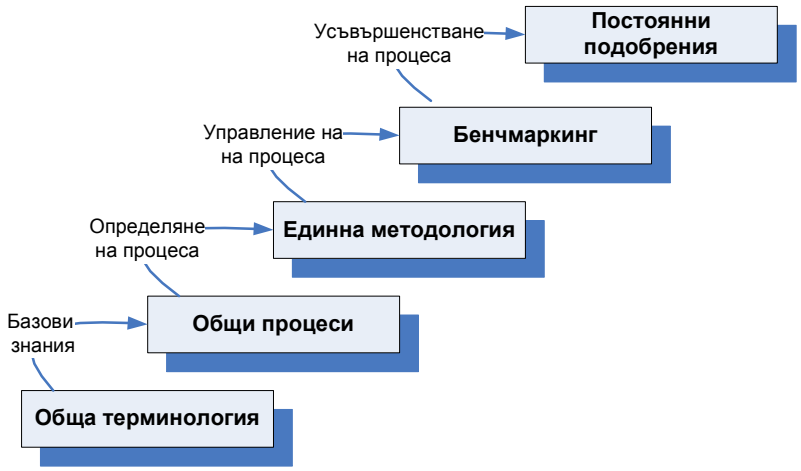
Категория ABE / DBB	**Преподавател**	**Студент**	**Дисциплина**
Същност	Преподавател	Студент	Дисциплина
Стратегия и план	Задачи и критерии		Задачи и критерии
Статуси	Научна степен, научно звание,	Образователен, семеен, криминален	Изпит
Комуникация	Комуникации с преподавателя	Комуникации със студента	Комуникации с други дисциплини
Конфигурация	Паспортни данни	Паспортни данни	Данни за дисциплината
Параметри за ефективност	Параметри за ефективност	Параметри за ефективност	Параметри за ефективност
Средства за тестване	Атестация	Изпити	Обсъждане
Проблеми	Проблеми на преподавателя	Проблеми на студента	Проблеми на дисциплината
История/биография	Биография на преподавателя	Биография на студента	История на дисциплината

3. Зрялост на градивните блокове

Отчитане нивото на зрялост на архитектурните компоненти при моделиране на данните е в основата на получаване детайлна и ясна картина за състоянието на архитектурата на предприятието и в частност на данни градивни блокове. Моделите на зрялост [18,19,20] определят визията и пътната карта за усъвършенстване на процеса по моделиране на данни в една организация.

3.1. Нива на зрялост

Моделът на зрялост на Гаролд Керцнер [3], известен като Project Management Maturity Model (PMMM) се състои от 5 нива (фиг.5), всяко от които представя различна степен на зрялост на процесите по създаване на модели на данни в една организация.



Фиг. 5. Нива на зрялост на градивните блокове

3.1.1. Обща терминология - организацията осъзнава важността от управление на данните и си поставя цели по тяхното моделиране [6, 7, 9] и последяващо пълноценно

използване. Започват проучвания и дейности по изучаване и получаване на базови знания за моделиране на данни и в частност за данни градивните блокове.

3.1.2. Общи процеси - организация е натрупала знания в областта на моделиране на данни и осъзнава необходимостта от общи процеси по управление, съхранение и поддържане качеството на данните, гарантиращи стандартизация на дейността и многократно използване [13].

3.1.3. Единна методология - организация осъзнава важността от използване на единна методология, базирана на архитектурния подход [11,15,19], гарантираща синергетичен ефект, от едновременната разработка на архитектурите на бизнес процесите, данните, приложенията и инфраструктурата.

3.1.4. Бенчмаркинг - на това ниво организацията проучва най-добрите практики в областта на информационното моделиране [7] и внедрява тези от тях, които съответстват на нейните цели и задачи.

3.1.5. Постоянни подобрения - на това ниво организацията е създала и поддържа добро ниво на управление на данните си. Периодично се проверява качеството и пълноценното им използване.

За модела са валидни следните правила:

- всяко следващо ниво започва след завършване на предходното;
- нивата могат да се припокриват;
- за всяко ниво се разработват показатели, обхващащи петте перспективи на рамката.

3.2. Критерии за зрялост

Критериите за зрялост дават представа за характеристиките на които трябва да отговарят данни градивните блокове и за тяхното влияние върху възможностите за преминаване към следващи нива на зрялост. Те могат да получат следната класификация :

3.2.1. Коректност - определя дали DBB съответства на правилата и техниките за моделиране на данни [9,15] (т.е. дали е валиден модел на данните). Това включва диаграми конвенции, правила за именуване, правила за структуриране, правилата за състава и нормализиране.

3.2.2. Пълнотата - отнася се до това дали DBB съдържа цялата информация, необходима за подкрепа на изискваната функционалност на системата в ABE.

3.2.3. Почтеност - определя дали DBB, включва всички бизнес правила, които се прилагат към данните.

3.2.4. Простотата - означава, че DBB съдържа минималните възможни същности и отношения.

3.2.5. Гъвкавостта - определя се лекотата, с която DBB може да се справя с промени в бизнеса или в нормативната уредба.

3.2.6. Интеграция - определя се като съвместимост на DBB с останалите архитектурни компоненти на модела на данни на организацията.

3.2.7. Разбираемост - определя се като лекота, с която концепции и структури в данни градивния блок мотат да бъдат разбрани.

3.2.8. Приложимост - определя се от това дали конкретния данни градивен блок, фигурира в хранилището на данни, както и може ли да бъдат изпълнен в рамките на срока, бюджетни ограничения и технология на проекта.

3.2.9. Стандартизираност - отразява доколко всички приложими стандарти са спазени, независимо от техния произход.

3.2.10. За многократна употреба - DBB трябва да осигурява възможност за многократна употреба, както самостоятелно, така и в състава на други по-мащабни модели.

3.2.11. Автоматизирана обработка - определя степента на формализация, която позволява да се използват програмни инструменти за въвеждане, съхранение, обработка, анализ и визуализация на описанието.

4. Заключение

Представеният материал дава най-обща представа за същността, използваните принципи и характеристиките на една много обсъждана в ИТ средите тема, каквато е архитектурата на данните [9, 15, 18, 19]. Направени са изводи за фундаменталната роля и мястото на архитектурата на данни в архитектурата на предприятието. Разглежда се мястото и ролята на важен, според автора, архитектурен елемент - данни градивни блокове. Представени са моделите, в които той участва, съдържанието на описанието му, както и примери за приложението му.

Разработена е таксономия на данни градивните блокове, която се базира на три основни класификационни признака: абстракция, обхват и зрялост. Тя може да послужи при специфициране и съхранение на информация за данни градивните блокове. Разработката може да се прилага като ръководен материал за ИТ отдели, които започват да реализират проекти в областта на синхронизацията на бизнеса и информационните технологии.

Литература

[1]. International Telecommunication Union , Enhanced Telecom Operations Map (eTOM) – The business process framework, printed in Switzerland, Geneva, http://www.billingcollege.com/upload/M.3050.1.pdf, 2005

[2]. International Telecommunication Union, Shared information and data model (SID, ITU-T M-Series Recommendations, Printed in Switzerland, Geneva, http://www.itu.int/rec/T-REC-M.3190-200807-I, 2009

[3]. Kerzner H. R, Using the Project Management Maturity Model: Strategic Planning for Project Management, John Wiley & Sons Inc, 2005

[4]. Lawrence A. T.,Taxonomy for the Technology Domain, Information Science Publishing (an imprint of Idea Group Inc.), 2005

[5]. Lubyansky A., Using Data Model Patterns to Build High-Quality Data Models Analyst, Project Performance Corporation-Part of the AEA group, http://www.ppc.com/assets/pdf/white-papers/Data-Model-Patterns.pdf

[6]. Milev V, H. Tujarov, S. Kalchev, Data PER model , International Conference on Computer Systems and Technologies - CompSysTech'13, 28-29 June 2013-Ruse, Bulgaria ,,2013

[7]. Milev V, S. Kalchev, Extended Information Model (EIM), http://www.tuj.asenevtsi.com/Public/EIM_UPLOAD.pdf

[8]. Milev V, S. Kalchev, Method For Data Modeling With Per Model, International Conference on Information Technologies (InfoTech-2013), 15-16 September 2013 Bulgaria, 2013

[9]. Milev V., The "Building Blocks" Concept Using for Modeling Data, International Conference on Information Technologies (InfoTech-2011), 15-16 September 2011 Bulgaria, 2011

[10]. Milev V., Multiple Dimensions Of Enterprise Architectural Description, Научна конференция с международно участие "Предизвикателства пред висшето образование и научните изследвания в условията на криза", Бургас 2010

[11]. Milev V., S. Kalchev, M. Stefanova, Architectural approach, based synchronization business and information technology, Научна конференция с международно участие "Mathtech2010"-Шумен 2010

[12]. TM Forum, Information Framework (SID), 2011, http://www.tmforum.org/InformationFramework/1684/home.html13

[13]. Tujarov H., S. Kalchev, V. Milev, Component model of the enterprise architecture, Proceedings of the International Conference on Information Technologies (InfoTech-2010) 16-18 September 2010, Bulgaria

[14]. Tujarov H., S. Kalchev, V. Milev, Methods for creation of Enterprise Architecture, International Conference on Information Technologies (InfoTech-2011), 15-16 September 2011 Bulgaria, 2011
[15]. Tujarov H., S. Kalchev, V. Milev., Using The “Building Blocks“ For Modeling Data, Международна научна конференция “Информатиката в научното познание 2010”, Варна 2010
[16]. Милев В., “3D модел на архитектура на данните”. "Приложение на ИКТ в икономиката и образованието", 2-3 декември, 2011, УНСС, София.”16
[17]. Милев В., В. Монев, Използване на таксономия при разработка на архитектура на данните, Международна научна конференция “Информационните технологии - стратегически приоритет в икономиката на знанието“, 14-15 октомври 2011г, Свищов, 2011
[18]. Милев В., С. Калчев, Архитектурата на данните като фундамент на архитектурата на предприятие Юбилейна международна научна конференция - 50 години ВТУ “ Св. св. Кирил и методий”, Велико Търново,2013
[19]. Тужаров Х., С. Калчев, В. Милев, Архитектура на предприятие, http://tuj.asenevtsi.com/EA/Index.htm, 2010
[20]. Тужаров Х., С. Калчев, В. Милев, Модел на зрялост на информационните технологии,международна научна конференция “Информационни технологии в управлението на бизнеса”, Варна 2009

За контакти:
докторант Васил Б. Милев
катедра „КСТ”
ВТУ „Св. св. Кирил и Методий“
E-mail: milev.vassil@gmail.com

Bulgaria Communications Chapter

МЕТОДИКА ЗА РАЗРАБОТКА НА РАЗШИРЕН ИНФОРМАЦИОНЕН МОДЕЛ

Стефан Калчев

Анотация: Значението на подходящ модел на данни и неговото управление е добре известно на IT общността. Тази статия има за цел да представи методика за използването на Разширен Информационен Модел (Extended Information Model, EIM), разработен от автора и включва:

- Кратко описание на модела.
- Речник на използваните термини.
- Дейности по реализацията на модела.

Ключови думи: архитектура на предприятия, архитектура на данните, моделиране на данни, , EIM, eTOM, SID, PER model.

Method for extended information model

Stefan Kalchev

Abstract: The importance of appropriate data models and their management is well known in IT community. This paper aims at presenting a method for using Extended Information Model (EIM) developed by he authors and includes:

- Brief description of the EIM model
- Glossary of terms
- Activities in realization of the method.

An example of the application of the model in particular problem domain, i.e. Education is given.

Keywords: enterprise architecture, data architecture, data modeling, EIM, eTOM, SID, PER model.

1. Въведение

В съвременния свят и особено във високотехнологичните компании нараства ролята на информацията, нейното структурно представяне и ефективното управление на данните. Разработеният от автора Разширен Информационен Модел (Extended Information Model, EIM) [8] осигурява на компаниите конкурентно преимущество, заключаващо се в:

- Реализация на прозрачен и надежден механизъм за управление на данните;
- Минимизиране на разходите свързани с администриране на голямо количество разпръснати и слабо интегрирани информационни системи;
- Повишаване на коефициента на използване на ИТ инфраструктурата;
- Оптимизиране на голямо количество параметри на операционната и стратегическа дейност на организацията.

Цел на настоящия доклад е да представи методика [17] за използване на EIM [8] модела в дейността на една организация. Методиката включва:

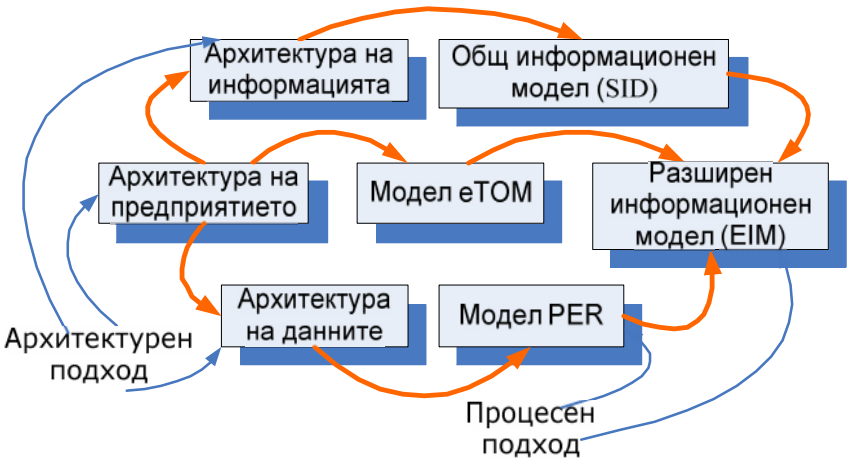
- Кратко описание на EIM модела;
- Речник на използваните термини;
- Дейности по реализация на методиката.

2. Разширен информационен модел

2.1. Основни принципи на модела

Моделът EIM служи за структуриране данните на цяла организация [12] и от тази гледна точка съответства на типа корпоративни модели на данните. Той представлява и официализира нещата, които са важни за една организация, както и правилата, които ги уреждат.

- Моделът е предназначен за организации с различна сфера на дейност, които се характеризират с ориентация към клиента и високо технологично ниво на оборудване.
- Моделът е създаден на достатъчно високо ниво на абстракция, което го прави приложим в организации с различна структура.
- Моделът представлява единно интегрирано представяне на данните, независимо от вида на системата или приложението. Независим от особеностите на практическата реализация модел, който служи за представяне на важни от гледна точка на бизнеса концепции и същности, техните характеристики и отношения.
- Моделът притежава структура на нива. Всяко ниво на модела EIM представлява конкретна архитектура, състояща се от няколко блока, построени по функционално-смислен признак.
- Моделът предвижда средство за визуализация, в качеството на което е избран моделът за моделиране на данни PER (Process-Entity-Relation). Диаграмите, построени в съответствие с PER модела [10], са нагледни и информативни. PER моделът [7], който е процесно ориентиран, улеснява взаимодействието с модела eTOM (The enhanced Telecom Operations Map) [3,6] и по точно с идентифицираните от него процеси.
- Моделът използва единна логика за представяне на данните във вид на агрегирани бизнес същности (Aggregated Business Entity, ABE) [8] и съставляващи ги данни градивни блокове [19]. Това подпомага задълбоченото разбиране на модела EIM и принципите на неговото създаване. Създава се основа за разработка на база EIM на собствени модели, предвиждащи цялостен поглед върху информационната среда на организацията.
- Моделът не се явява изолирана структура, което е едно от неговите съществени предимства. Той се базира на три модела (фиг. 1):
    - Модел eTOM [3]– разработка на карта на процесите, които се декомпозират до 4-то ниво.
    - Модел SID (Shared information and data model) [4,6,15,20] - информационен модел на организацията, на който се базира разработеният от авторите модел EIM.
    - Модел PER [7,10] - модел за структуриране на данните, в който обектите са типизирани и разположени на йерархични нива според предназначението си.



Фиг. 1. Произход на EIM

2.2. Базови компоненти на модела EIM

На следващата фиг. 2 са представени базовите компоненти на модела EIM.

Фиг. 2. Базови компоненти на EIM

Модел на предметната област

Целта на модела на предметната област е да улови и представи съответните бизнес изискванията. За такъв модел е важно да бъде прост и лесен за разбиране. Този модел на данните ще представлява основа за по-нататъшен анализ, така че не е необходимо да улови всички детайли. В EIM моделът се използва за определяне на основните бизнес данни, с които работи организацията, наречени агрегирани бизнес същности (ABE) [8,12].

Концептуален модел

Обикновено се създава в етапа стратегия за управление на информацията. Той съдържа ключовите обекти и отношения и представя от високо ниво структурата на данните в рамките на една организация.

Агрегирани бизнес същности

Основни данни на организацията, структурирани като тясно свързани функционално различни обекти, обединени в една група [8,12]. Примери: клиент, стока, ресурс и т.н.

Данни Градивни Блокове

Описват конкретни архитектурни решения в областта на моделиране на данни, които се явяват типови и могат многократно да бъдат използвани в различни приложения.

Представляват функционално обособени един или група обекти, които реализират конкретна функция в рамките на домейн от конкретна предметна област [11,19]. Примери: Клиенти основни данни, клиенти фирмени данни, комуникация с клиенти и т.н.

3. Речник

Архитектурен подход [9] - идеята за тясно и ефективно взаимодействие на бизнеса и информационните технологии става основа на архитектурния подход, при който тези две понятия се разглеждат като едно цяло - архитектура на предприятието. В основата на синхронизацията на бизнеса с информационните технологии стои архитектурата на данните, която пък се базира на информационните модели на данните.

Процесен подход [21] - дейността на всяка организация представлява верига от свързани помежду си бизнес процеси - от маркетинга и планирането до продажбите и обслужването след тях. Естествено е да се търси връзка между веригата свързани процеси и обезпечаващите ги данни при реализиране на информационни системи.

Архитектура на информацията [16,17] **-** процес на организация и представяне на значима за потребителите информация в интуитивно разбираема форма, с използване на съответни средства за каталогизация и навигация.

Архитектура на данните [18] - състои се от модели, политики, правила и стандарти, които регламентират кои данни се събират и как се съхраняват, подредени, интегрирани и използвани в информационните системи и в организациите.

Информационен модел [20] - отразява процесите по възникване, предаване, преобразуване и използване на информация в системи с различна природа.

Обект (същност) - множество екземпляри, притежаващи едни и същи характеристики и подчиняващи се на едни и същи правила. Всеки обект в информационния модел трябва да има уникално име и да бъде идентифициран с ключ. Следващите обекти се явяват основни компоненти на модела PER [7].

Обект връзка (ОВ) - логическо свързване между два или повече типови обекта. Използва се при релации от тип “много към много”. За идентифициране използва съставен ключ.

Обект процес (ОП) - включва данни за процеса, предмет на моделиране. Изразява се с глагол - продава, купува, поръчва.

Основен обект (ОО) - включва основни данни за обекти, които пряко или косвено участват в процеса. Изразява се със съществително име -. клиенти, стоки, книги, автори.

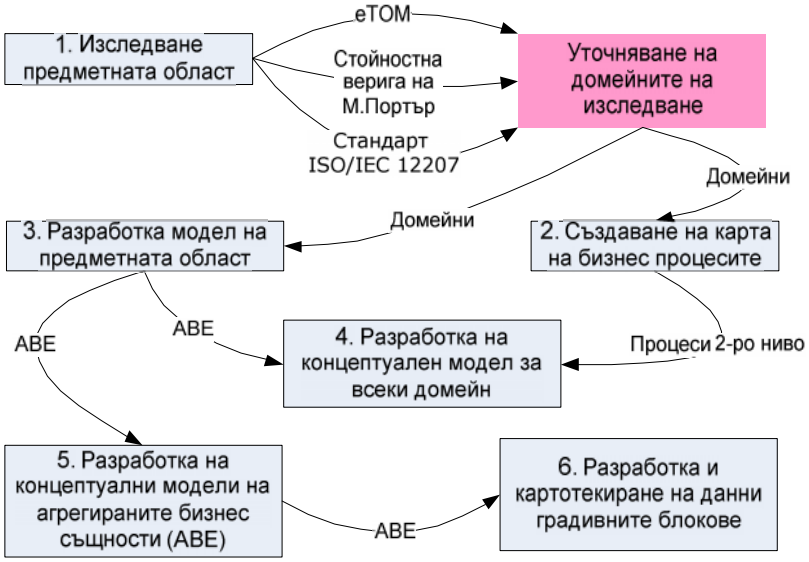
Подчинен обект (ПО) - представя подробни данни за основния обект. Изразява се с името на основния обект и пояснение - клиент ФД (фирмени данни), клиент КОМ (комуникации), клиент ИД (индивидуални данни).

Спомагателен обект (СО) - вид подчинени обекти, които съществуват относително самостоятелно в рамките на приложението и не участват в обекта процес. Те могат да бъдат напълно самостоятелни или обвързани с основните обекти в релация n:m - автори на книги, събития със студента и т.н.

Обект списък (ОС) - включва данни с относително постоянен характер, явяващи се атрибут на основния обект. Изразява се със съществително - град, професия, социален статус, образование, отдел.

4. Методика за реализация на модела

Методиката описва дейностите (фиг. 3), които е необходимо да бъдат реализирани при разработка на модел EIM за една организация. Като предметна област за демонстрация на методиката е избрана областта ВУЗ – висше учебно заведение.



Фиг. 3. Дейности за реализация на модела EIM

4.1. Изследване на предметната област

В зависимост от спецификата на предметната област се използва един от предложените по-долу подходи за изследване дейностите в една организация:

- **Според модела eTOM** [3] **-** схемата eTOM в графичен вид илюстрира бизнес-процесите, необходими за функциониране на телеком оператора на услуги. На нулево ниво на eTOM диаграмата са разположени три вида процеси:
  - **Стратегия, инфраструктура и продукти** (SIP - Strategy, Infrastructure, Product) - включва целия спектър въпроси, свързани със стратегията на организацията, развитието на инфраструктурата и управление на жизнения цикъл на продуктите (услугите).
  - **Операционни процеси** (Operations) - явяват се най важните процеси в една телекомуникационна компания, основа на които се явява представянето и осигуряването на услуги и биллинг.
  - **Управление на организацията** (Enterprise Management) - обхваща общи въпроси по осигуряване дейността на организацията, такива като управление на кадри, финанси и активи, знания, външни връзки и т.н.

- **Според стойностната верига на М. Портър** [13,14] **-** стойностната верига е от полза при анализиране на определени дейности, чрез които организациите могат да генерират стойност и конкурентно предимство. Организацията се представя като верига от дейности, създаващи стойност. Портър разграничава поредица от чести, взаимно свързани генерични дейности във фирмите. Полученият модел е известен като стойностна верига, където Портър определя основни и поддържащи дейности, както е показано на фиг. 4.

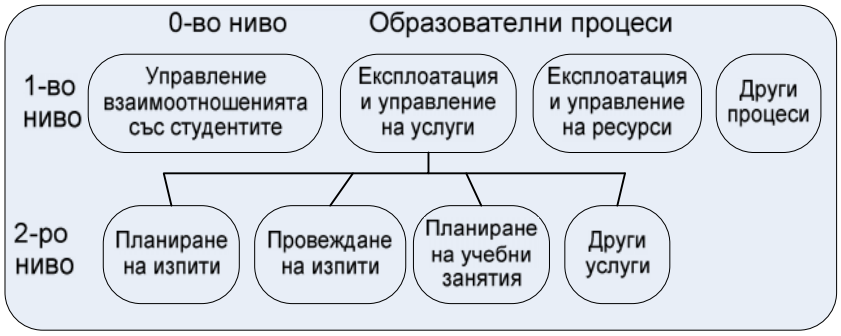
Фиг. 4. Стойностна верига на М. Портър

- **Според Стандарта ISO/IEC 12207** (фиг. 5) [2, 5] **-** Стандартът описва процесите, които се извършват при разработка на програмно осигуряване.

Фиг. 5. Видове дейности според Стандарта ISO/IEC 12207

4.2.Създаване на карта на бизнес процесите eTOM

За всеки от избраните по-горе домейни се разработва карта на бизнес процесите. На фиг. 6 са представени образователни процеси с декомпозиция на образователните услуги до второ ниво.



Фиг. 6. Образователни процеси

На второ ниво на декомпозиция съответният процес се реализира във вид на модел на данните.

4.3. Разработка модел на предметната област

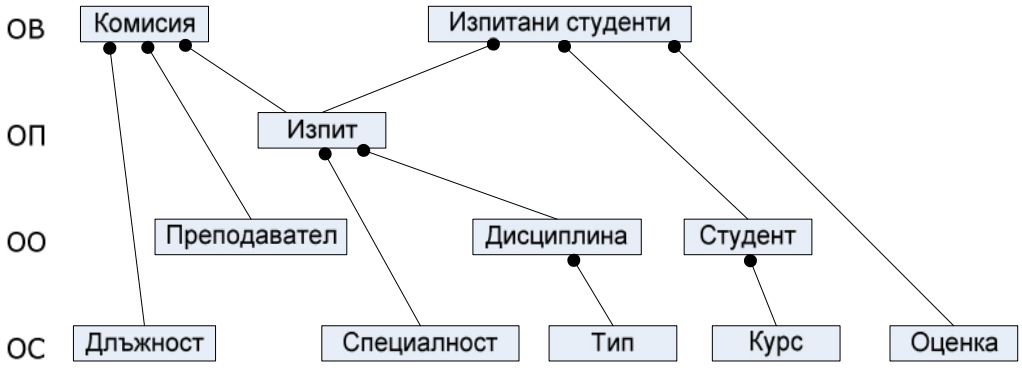
Моделът на предметната област представя в груб и опростен вид основните обекти в една предметна област и връзките помежду им. Той се описва с помощта на стандарта CDIF (CASE Data Interchange Format) [1].

Фиг. 7. Модел на предметната област ВУЗ

От него се извеждат основните данни на организацията, представляващи АВЕ. За предметната област ВУЗ това са: преподавател, студент, учебни услуги, дисциплина, ресурс и т.н.

4.4. Разработка на концептуален модел на всеки домейн

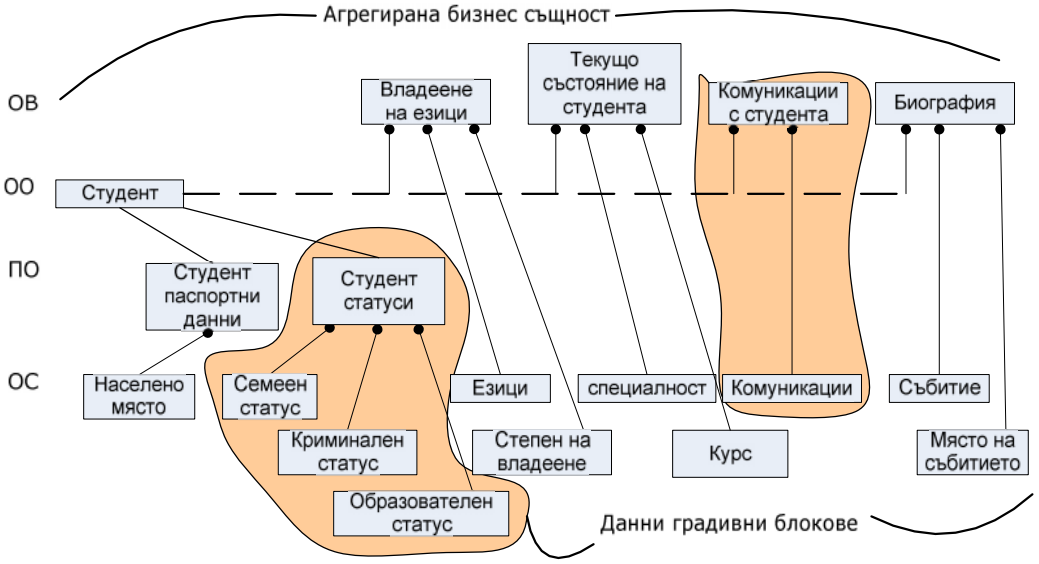
Моделът EIM предвижда разработката на два вида концептуални модела - процес ориентиран и агрегирана същност ориентиран. На тази фаза от методиката се разработват процес ориентираните концептуални модели на данни, които съответстват на 2-ро ниво на декомпозираните процеси в модела eTOM. Пример за провеждане на изпит е даден на фиг.8.



Фиг. 8. Концептуален модел на изпит

4.5. Разработка на концептуални модели на агрегирани бизнес същности

АВЕ са най-важният архитектурен компонент в модела EIM. Чрез тях се събират пълни данни за конкретна бизнес единица. На фиг. 9 е представена "АВЕ – Студент".



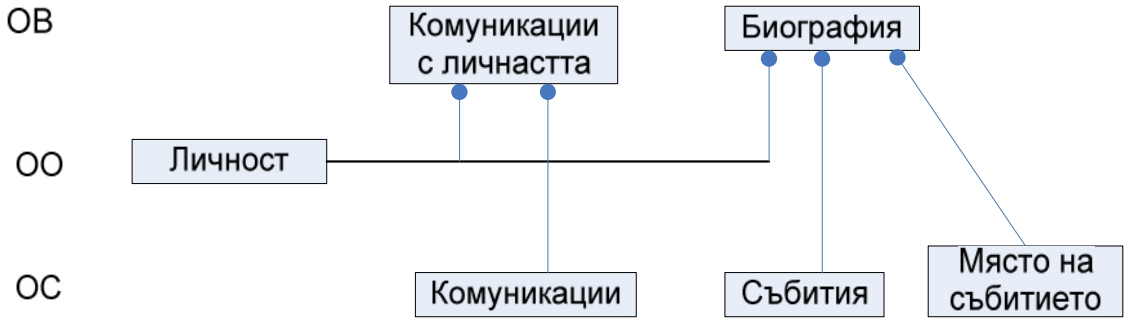
Фиг. 9 . ABE - Студент

Те се изграждат на базата на компонента данни градивни блокове [11,18]. В състава им е включена базова същност (в случая - студент), чрез която те участват в процес ориентирания компонентен модел. За описанието на ABE се използват следните данни:

- Наименование;
- Предназначение;
- Съдържание - включените данни градивни блокове;
- Участие - бизнес процесите, в които участва ABE.

4.6. Разработка и картотекиране на данни градивните блокове

Данни градивните блокове са типови архитектурни елементи, които се използват многократно при разработване на концептуални модели на ABE. Пример е даден на фиг. 10: Градивният блок “Комуникации със студента“ се използва при разработка на модел на ABE, отнасящ се до личност или организация. Градивния блок “Биография“ се използва при разработка на ABE, отнасящ се до личност



Фиг. 10. Данни градивните блокове - Комуникации и Биография

За описание на данни градивните блокове се използват следните данни:

- Извършва се текстово описание на същността;
- Изброяват се ABE, в които градивния блок участва;
- Изброяват се същностите които участват в градивния блок;
- Описват се правила на използване и реализация.

5. Заключение

В доклада е представена методика за разработка на модела EIM. Описани са основните принципи на модела, които го правят приложим за моделиране данните на една организация. Накратко са изяснени основните компоненти на модела. В речник са описани основните понятия, използвани в методиката. Дейностите в методиката са описани последователно със съответни примери.

Методиката може да послужи за структуриране данните на цяла организация [11,18,21] и от тази гледна точка да официализира нещата, които са важни за една организация, както и правилата, които ги уреждат.

Литература

[1]. Chen, M.,CASE data interchange format (CDIF) standards: Introduction And Evaluation, System Sciences, Proceeding of the Twenty-Sixth Hawaii International Conference on,1993

[2]. fkn+antitotal, Стандарт ISO/IEC 12207-95 , http://fkn.ktu10.com/?q=node/744

[3]. International Telecommunication Union , Enhanced Telecom Operations Map (eTOM) – The business process framework, printed in Switzerland, Geneva,
http://www.billingcollege.com/upload/M.3050.1.pdf, 2005

[4]. International Telecommunication Union, Shared information and data model (SID, , ITU-T M-Series Recommendations, Printed in Switzerland, Geneva, http://www.itu.int/rec/T-REC-M.3190-200807-I, 2009

[5]. ISO/IEC 12207:2008(E), Licensed to University of Padova / Dr. Vardanega
http://www.math.unipd.it/~tullio/IS-1/2009/Approfondimenti/ISO_12207-2008.pdf

[6]. Markovits S., M. Lam , R. Braun, Information Modeling of Trouble: A Service Provider View, University of Technology Sydney, Australia,
http://teleholonics.eng.uts.edu.au/pubs_archive/TT_repContel.pdf

[7]. Milev V, H. Tujarov, S. Kalchev, Data PER model , International Conference on Computer Systems and Technologies - CompSysTech'13, 28-29 June 2013-Ruse, Bulgaria , 2013

[8]. Milev V, S. Kalchev, Extended Information Model (EIM),
http://www.tuj.asenevtsi.com/Public/EIM_UPLOAD.pdf

[9]. Milev V., S. Kalchev, M. Stefanova, Architectural approach, based synchronization business and information technology, Научна конференция с международно участие "Mathtech2010"-Шумен ,2010

[10]. Milev V., S. Kalchev, Method For Data Modeling With PER Model, Proceedings of the International Conference on Information Technologies (InfoTech-2013) 19-20 September 2013,Varna, Bulgaria, 2013

[11]. Milev V., The "Building Blocks" Concept Using for Modeling Data, International Conference on Information Technologies (InfoTech-2011), 15-16 September 2011 Bulgaria, 2011

[12]. Pignatelli G., G. Motta, Strategic Modelling of Enterprise Information Requirements: a normative model of information domains and information types, Department of Computer Engineering and Systems Science, University of Pavia, Italy, 2008
http://camellia.unipv.it/wcc/download/SSME-GMM-v1.pdf

[13]. Porter M., Competitive Advantage: Creating and Sustaining Superior Performance , Free Press, 1998, ISBN: 0684841460

[14]. Porter M., Competitive Strategy, Free Press, 1980

[15]. TM Forum, Information Framework (SID),
http://www.tmforum.org/InformationFramework/1684/home.html13, 2011

[16]. Tujarov H., S. Kalchev, V. Milev, Component Model Of The Enterprise Architecture, Proceedings of the International Conference on Information Technologies (InfoTech-2010) 16-18 September 2010,Bulgaria, 2010
[17]. Tujarov H., S. Kalchev, V. Milev, Methods for creation of Enterprise Architecture, International Conference on Information Technologies (InfoTech-2011), 15-16 September 2011, Bulgaria, 2011
[18]. Милев В., С. Калчев, Архитектурата на данните като фундамент на архитектурата на предприятие Юбилейна международна научна конференция -50 години ВТУ “ Св. св. Кирил и методий”, Велико Търново, 2013
[19]. Милев В., Таксономия на данни градивни блокове, http://www.tuj.asenevtsi.com/Public, 2013
[20]. Тужаров Х., Методологии и стандарти за управление на телекомуникационни услуги, Общ информационен модел SID, http://www.tuj.asenevtsi.com/TMF/TMF14.htm 16,2011
[21]. Тужаров Х., С. Калчев, В. Милев, Архитектура на предприятие, http://tuj.asenevtsi.com/EA/Index.htm, 2010

За контакти:
гл. ас. Стефан Калчев
катедра „КСТ”
ВТУ „Св. св. Кирил и Методий“
E-mail: stkalchev@mail.bg

Bulgaria
Communications
Chapter

СПЕКТРАЛЕН АНАЛИЗ НА ПОСЛЕДОВАТЕЛНОСТИТЕ ПОЛУЧЕНИ ОТ ОБОБЩЕН Р-ИЧЕН САМОСВИВАЩ ГЕНЕРАТОР

Антония Т. Ташева

Резюме: Статията представя Фурие анализ на последователностите, получени от обобщен р-ичен самосвиващ генератор. Представени са резултатите от спектралния анализ над тези последователности, доказващи практически шумоподобните свойства на генератора и възможността за неговото приложение на практика.
Ключови думи: pGSSG, криптография, статистика, псевдослучайни последователности

Spectral analysis of the sequences obtained by the p-ary Generalized Self Shrinking Generator

Antoniya T. Tasheva

Abstract: The paper introduces the Fourier analysis on the sequences of the *p*-ary Generalized Self Shrinking Generator. The results of the spectral analysis of those sequences are presented, proving the practical randomness properties of the generator and its ability to be used in practice.
Keywords: pGSSG, Cryptography, Statistics, Pseudorandom sequences

1. Увод

Технологичният напредък на 21-ви век неминуемо налага нуждата от сигурност на информацията, предавана по „мрежата“. За целта в последните десетилетия се развиват изследванията в областта на криптографията, като те са насочени към създаване, обосноваване и изследване на нови начини за криптиране на информацията. Широко приложение в тази област намират поточните шифри, основаващи се на простата операция сума по модул две, приложена върху явния текст и ключова последователност от псевдослучайни битове. Те предоставят възможност за криптиране в реално време и за да бъдат реално приложими на практика, трябва да бъдат изследвани и доказани техните статистически свойства и устойчивост на криптоанализ.

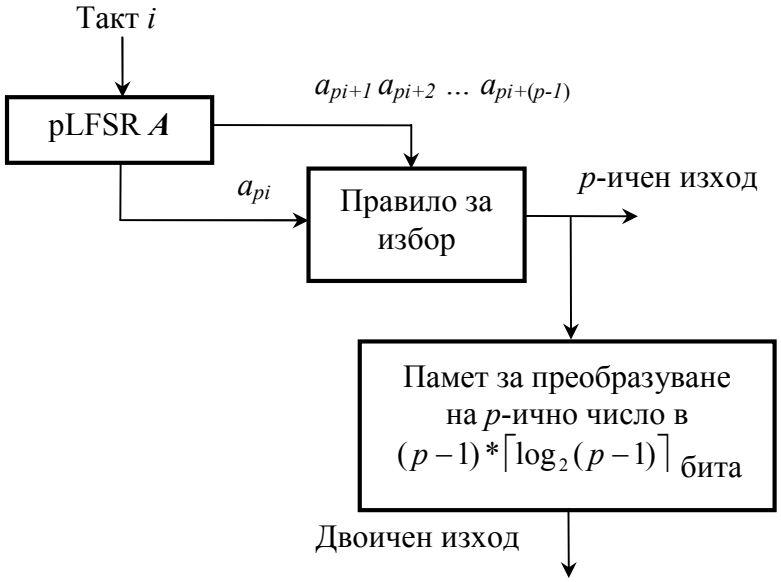
Този доклад е посветен на проведения спектрален анализ върху последователности, получени от обобщения р-ичен самосвиващ генератор, предложен в [6]. В началото доклада представя накратко архитектурата на генератора. Следва описание на принципа на спектралния анализ и неговия алгоритъм. Накрая са поместени резултатите и изводите от тях.

2. Същност на обобщения р-ичен самосвиващ генератор

Обобщеният *p*-ичен самосвиващ генератор pGSSG (*p*-ary Generalized Self-Shrinking Generator) представлява модификация на самосвиващия генератор SSG (Self-Shrinking Generator) на Майер и Щефелсбах [3] и за разлика от него, работи за всяко произволно просто число *p*, а не само при $p = 2$. На фигура 1 е представена архитектурата на pGSSG.

Архитектурата на *p*-ичния обобщен самосвиващ генератор pGSSG се състои от един *p*-ичен LFSR (pLFSR - *p*-ary Linear Feedback Shift Register) регистър A с дължина L (фиг. 1). Той генерира *m*-последователност $(a_i)_{i\geq 0}$ от *p*-ични цифри (т.е. $(a_i)_{i\geq 0}, 0 \leq a_i \leq p-1$) и

$0 \le i \le L-1$. Множителите на обратните връзки в pLFSR се определят от коефициентите на примитивен полином в разширението на полето на Галоа GF(p^L). Всеки градивен елемент на pLFSR може да съхранява една p-ична цифра. Регистърът се инициализира с p-ичната последователност $(a_0, a_1, \ldots, a_{L-1})$.



Фиг. 1. P-ичен обобщен самосвиващ генератор

Част от изходната pLFSR последователност се избира и за изход на pGSSG посредством описания в долното определение алгоритъм.

***Определение*:** Алгоритъмът на p-ичния обобщен самосвиващ генератор се състои от следните стъпки:

1. P-ичният LFSR се тактува с тактова последователност с период T.
2. Изходната pLFSR последователност се разделя на p-торки $\left(a_{pi}, a_{pi+1}, a_{pi+2}, \ldots, a_{pi+(p-1)}\right)$, $i = 0, 1, \ldots$
3. Ако $a_{pi} = 0$, то цялата p-торка се отхвърля, т.е. изходната pGSSG последователност е свита.
4. Когато $a_{pi} \neq 0$, съответната цифра $a_{pi+a_{pi}}$ от p-торката формира част от изхода на GSSG.
5. Самосвитата p-ична GSSG изходна последователност се преобразува в двоична като всяка p-ична цифра се представя с $\lceil \log_2(p-1) \rceil$ бита, където $\lceil x \rceil$ е най-малкото цяло число, по-голямо или равно на x. Всяка изходна цифра i със стойност от 1 до $p-1$ от p−ината GSSG последователност се представя чрез двоичното представяне на числото:

$$(i-1) + \frac{2^{\lceil \log_2 p-1 \rceil} - p - 1}{2} \qquad (1)$$

Като пример, преобразуването на всяка p-ична цифра за p = 3, 5, 7, 11, 13 е показано в Таблица 1.

6. Всяка p-ична нула при i-тата си поява $(i = 1, 2, 3, \ldots)$ в генерираната p-ична последователност се представя чрез двоичното представяне на числото d_i,

$$d_i = \begin{cases} (d_{i-1} + 1) \bmod p, & \text{ако } d_{i-1} < p - 1 \\ 1, & \text{ако } d_{i-1} = p - 1 \end{cases} \quad (2)$$

при начално състояние $d_0 = 0$.

От алгоритъма на работа на pGSSG генератора следва, че генерираната p–ична псевдослучайна последователност е *свита, смалена* версия на pLFSR последователността, когато стойността на първата цифра в p-торката е нула. В противен случай, когато стойността на първата цифра в p-торката е различна от нула, в генерираната последователност се извежда съответната цифра в p-торката.

Таблица 1. Двоично преобразуване на p-ичната цифра.

***p*-ична цифра**	**Двоично представяне на *p*-ичната цифра**				
	***p* = 3**	***p* = 5**	***p* = 7**	***p* = 11**	***p* = 13**
1	0	00	001	0011	0010
2	1	01	010	0100	0011
3	-	10	011	0101	0100
4	-	11	100	0110	0101
5	-	-	101	0111	0110
6	-	-	110	1000	0111
7	-	-	-	1001	1000
8	-	-	-	1010	1001
9	-	-	-	1011	1010
10	-	-	-	1100	1011
11	-	-	-	-	1100
12	-	-	-	-	1101

Основен елемент в архитектурата на pGSSG генератора е недвоичният pLFSR регистър. Неговите алгебрични свойства са обобщени в разработките на Golomb и Gong [1], [2]. За сега той се използва по-рядко в сравнение с двоичните LFSR регистри.

3. Основи на спектралния анализ

За да се открие дали генерираните от pGSSG последователности имат периодични свойства е необходимо да се извърши спектрален анализ. Това е един от най-важните статистически тестове, защото откриването на периодичност индикира сериозни отклонения от случайните свойства на изследвания псевдослучаен генератор. Спектралният тест изследва пиковите стойности в дискретното преобразование на Фурие на последователността, генерирана от псевдослучайния генератор. Целта му е да провери дали броят на пиковете в дискретното преобразование на Фурие, по-големи от 95% от определена прагова стойност, значително се различава от 5 % от всички стойности.

Фурие анализът се състои от следните етапи:

1. Логическите нули „0” и единици „1” от изходната последователност $\boldsymbol{A} = (a_1, a_2, \ldots, a_n)$ на *pGSSG* генератора се преобразува в последователност $\boldsymbol{X} = x_1, x_2, \ldots, x_n$ от „–1” и „+1”, където $x_i = 2a_i - 1$, $i = 1, \ldots, n$.

За извършване на теста се препоръчва да се изследват последователности с дължина минимум 1000 бита.

2. Извършване на дискретно преобразование на Фурие (ДПФ) над последователността ***X.***

$$f_j = \sum_{k=0}^{n-1} X_i \left[\cos\left(\frac{2\pi kj}{n} \right) + i \sin\left(\frac{2\pi kj}{n} \right) \right], \qquad j = 0, \ldots, n-1 \tag{3}$$

Получената последователност ***F*** от комплексни променливи представя периодичните компоненти на последователността от битове за различни честоти.

3. Намиране на последователността ***M*** от модулите на комплексните честоти f_j.

Поради симетрията на ДПФ само стойностите от 0 до $n/2 - 1$ се разглеждат $\boldsymbol{M} = |\boldsymbol{F'}|$, където ***F´*** е последователност от първите *n/2* елемента от ***F***. Последователността ***M*** представлява редицата от пикови стойности.

4. Изчисляване на праговата стойност $T = \sqrt{3n}$.

За да удовлетворява изискванията за случайна псевдослучайна последователност, 95 % от пиковите стойности не трябва да превишават прага *T*.

5. Изчисляване на очаквания теоретичен брой N_0 от пиковите стойности, които са по-малки от прага *T*

$$N_0 = 0.95n/2. \tag{4}$$

6. Преброяване на реалния брой пикове N_1 от последователността ***M*** под прага *T*.

7. Изчисляване на стойността

$$d = \frac{N_1 - N_0}{\sqrt{\frac{n.0{,}95.0{,}5}{2}}} \tag{5}$$

8. Изчисляване на *P–стойността*

$$P - стойност = erfc\left(\frac{|d|}{\sqrt{2}} \right) = \frac{2}{\sqrt{\pi}} \int_{|d|/\sqrt{2}}^{\infty} e^{-u^2} du \; . \tag{6}$$

9. Оценка на резултата от Фурие анализа.

Ако изчислената *P–стойност* < 0.01, то последователността не е псевдослучайна. В противен случай последователността може да се счита за случайна.

4. Експериментални резултати

За провеждането на тестовете е използвана методиката, предложена от Националния институт по стандартизация и технологии на САЩ (NIST - National Institute of Standards and Technology) [4, 5]. Тя съдържа множество тестове, които са предназначени за откриване на отклонение на поведението на тестваните двоични последователности от напълно случайните такива. NIST препоръчва използването на тези тестови процедури като първа стъпка в проверката на приложимостта на псевдослучайните генератори (PRNG – Pseudo Random Number Generators) за специфичните криптографски приложения.

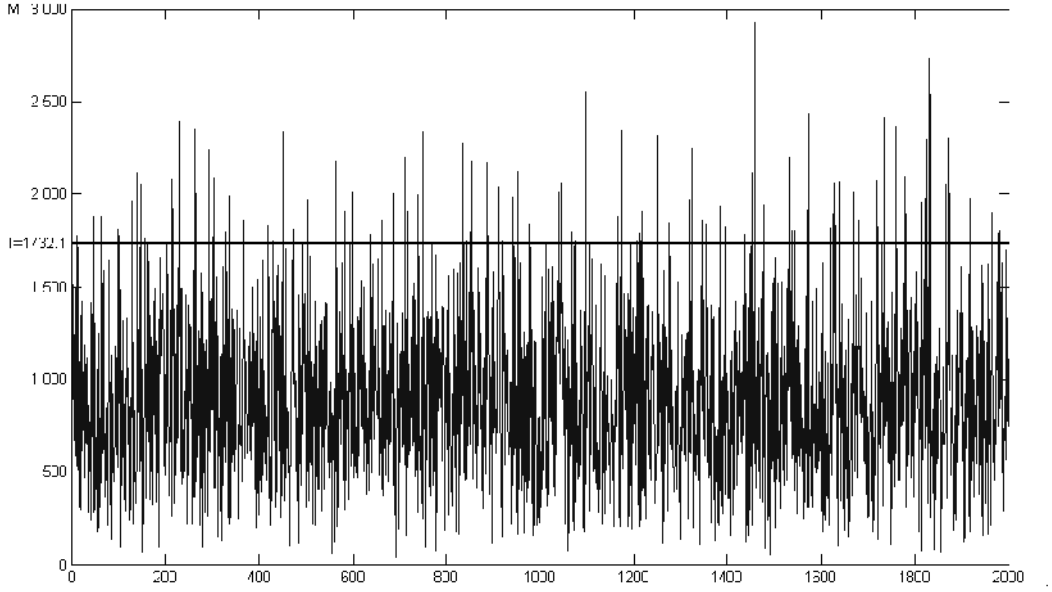
За целта на провеждането на спектралния анализ са тествани 3 различни *pGSSG* генератора с полиноми на обратните връзки, дадени в таблица 2. Извършени са по 100 теста върху последователности с дължина 1 000 000 бита, в резултат на което са получени 300 различни *P–стойности*.

Таблица 2. Полиноми на връзките на използваните pGSSG

№	***pGSSG/LFSR***	**Полином на обратната връзка**
1	*LFSR, p = 257, L = 32*	$x^{32} + x + 10$
2	*LFSR, p = 257, L = 32*	$x^{32} + 75\,x^2 + 174\,x + 33$
3	*LFSR, p = 257, L = 32*	$x^{32} + 188\,x^2 + 200\,x + 107$

На фиг. 2 са показани модулите ***M*** на първите 2 000 честотни компоненти от ДПФ на последователност с дължина 1 000 000 бита, генерирана от 1[ви] *pGSSG*. С плътна линия е представена теоретичната прагова стойност ***T* = 1732,1**. Фигурата потвърждава практически получения резултат, че по-малко от 5% (49674 от 1 млн.) от пиковите стойности попадат над прага ***T***. Полученият спектър е шумоподобен, което индикира факта, че не съществуват периодично повтарящи се компоненти в последователностите, генерирани от *pGSSG*.

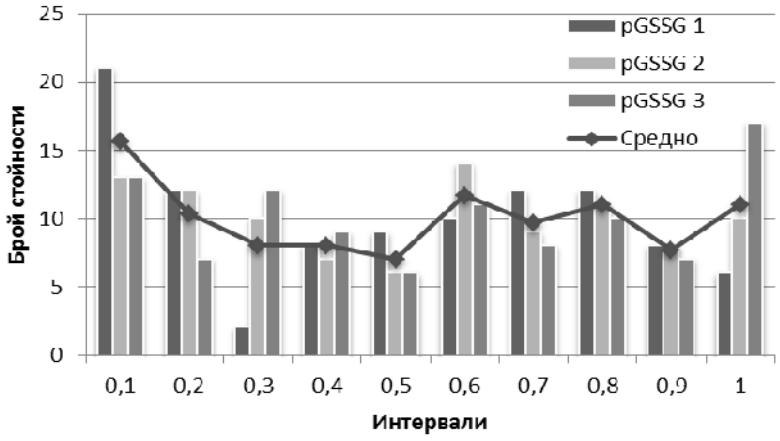
При извършване на статистическия Фурие анализ е избрано ниво на значимост α = 0.01. Средните стойности на резултатите от Фурие анализа на изследваните 300 изходни последователности са представени в таблица 3, а хистограмата на разпределение на *P–стойностите* е представена на фиг. 3.



Фиг. 2. Модули на честотните компоненти от ДПФ на pGSSG последователност

Таблица 3. Резултати от Фурие анализа на pGSSG

Средна стойност	***P*-стойност**	**Пропорция**
от 100 теста на 1[ви] pGSSG	0,008266	97%
от 100 теста на 2[ри] pGSSG	0,739918	99%
от 100 теста на 3[ти] pGSSG	0,334538	97%
Обща средна стойност	**0,360907**	**97,67%**



Фиг. 3. Хистограма на P-стойностите от Фурие анализа

Тъй като *P-стойност* ≥ *0.0001* [5], последователността генерирана от *pGSSG* може да се смята за *непериодична с доверителна вероятност 99%*. Пропорцията (таблица 3) на последователностите, преминали спектралния тест, се определя от броя на последователностите, които имат *P-стойности* ≥ *0.01*. Изводът, който може да се направи е, че *всички пропорции попадат в доверителния интервал* 97,28% ÷ 100%, като всички 300 изследвани последователности *са преминали успешно спектралния Фурие анализ*.

От анализа на резултатите може да се направи изводът, че изследваните последователности са непериодични и равномерно разпределени и отговарят на поставените към тях изисквания за псевдослучайност.

5. Заключение

Направеният статистически Фурие анализ на генерираните *pGSSG* последователности доказва, че техният спектър е шумоподобен. Това показва, че в последователностите, получени от *pGSSG* генератора, не съществуват периодично повтарящи се компоненти, т.е. те са ***непериодични и равномерно разпределени***. Тези свойства потвърждават, че *pGSSG* последователностите са напълно непредсказуеми и независещи от ядрото на генератора.

Приложеният анализ на Фурие, както и проведените други статистически тестове [7], доказват шумоподобните свойства за изходните последователности на Обобщения p-ичен самосвиващ генератор, което дава основание да се твърди, че генераторът би имал практическо приложение. Това обаче е необходимо, но не и достатъчно условие за неговата безусловна приложимост, затова в бъдеще е необходимо да се извърши обстоен криптоанализ на предложения генератор.

Литература

[1]. Golomb S. W., Gong G., Signal Design for Good Correlation: For Wireless Communication, Cryptography, and Radar, Cambridge University Press, 2005.

[2]. Gong G., Sequence Analysis, University of Waterloo, 1999, p. 137, http://calliope.uwaterloo.ca/~ggong

[3]. Meier, W., Staffelbach, O., The self-shrinking generator, *Advances in Cryptology - EUROCRYPT '94*, 1995, pp. 205-214.

[4]. NIST Statistical Test Suite, Version 2.1.1., August 11, 2010, http://csrc.nist.gov/groups/ST/toolkit/rng/documentation_software.html

[5]. Rukhin, A., et.al, NIST Special Publication 800-22rev1a: A Statistical Test Suite for the Validation of Random Number Generators and Pseudo Random Number Generators for Cryptographic Applications, April 2010, http://csrc.nist.gov/groups/ST/toolkit/rng/documents/SP800-22rev1a.pdf.

[6]. Tasheva, A.T., Tasheva, Z.N., and Milev, A.P., Generalization of the self-shrinking generator in the Galois Field GF(p^n), *Advances in Artificial Intelligence*, 2011, Article ID 464971, 10 pages, doi:10.1155/2011/464971

[7]. Tasheva, A.T., Nakov, O., Trifonov, T., Statistical Analysis of the Sequences Produced by the p-ary Generalized Self-Shrinking Generator, Recent Advances in Computer Science, WSEAS Press, 2013

За контакти:
ас. инж. Антония Т. Ташева
катедра „Компютърни системи”
Факултет Компютърни Системи и Управление
Технически Университет - София
e-mail: atasheva@tu-sofia.bg

COMPUTER-MATHEMATICAL ALGORITHM FOR THE SOLUTION OF THE GENERAL SECOND PROBLEM OF PORTFOLIO SELECTION UNDER RISK-PART 1-THEORY

Lyubomir N. Sotirov, Stela L. Sotirova-Nikolova, Samuil V. Nikolov, Vladimir N. Nikolov, Trifon I. Ruskov, Ivan T. Ruskov

Abstract. The terms of strictly convex, convex and non-convex problems of portfolio selection under risk and new measures of risk are introduced. Algorithms for direct and iterative solving of the general second problem of portfolio selection under risk are proposed. The synthesized analytical solutions allow the interpretation of the investigated optimizational models of the expected return at fixed acceptable minimal risk as an information service in Internet.
Keywords. portfolio selection, strictly convex, convex and non-convex problems, expected return, risk, linear algebraic systems equations, information service in Internet

Компютърно-математически алгоритъм за решаване на обща втора задача за избор на портфейл в условия на риск - част 1 - теория

Любомир Н. Сотиров, Стела Л. Сотирова-Николова, Самуил В. Николов, Владимир Н. Николов, Трифон И. Русков, Иван Т. Русков

Резюме: Въвеждат се понятията строго изпъкнали, изпъкнали и неизпъкнали проблеми за избор на портфейл в условия на риск, както и нови начини за измерване на риска. Предлагат се алгоритми за директно и итеративно решаване на общата втора задача за избор на портфейл в условия на риск. Синтезираните аналитични решения позволяват интерпретирането на изследваните оптимизационни модели на очакваната възвръщаемост при определен приемлив минимален риск като информационна услуга в интернет.
Ключови думи: избор на портфейл, строго изпъкнали, изпъкнали и неизпъкнали проблеми, очаквана възвръщаемост, риск, системи линейни алгебрични уравнения, информационни услуги в интернет

1. Introduction

By investing funds in various securities, the investor creates an investment portfolio with the purpose of obtaining income in the future.

His goal is to form this portfolio in a certain way so as to minimize risk and to obtain the desired expected return or to maximize expected return at a determined risk level.

We will name these formulations a first and a second problem of Harry Markowitz [5,6,7]. We will investigate the more general case of second problem of portfolio selection, where optimizational model is of the expected return at fixed acceptable minimal risk.

Obviously, the first problem can be solved by the methods of strictly convex, convex and non-convex quadratic programming under linear limitations, investigated by the authors as well (see references).

The algebraic method, proposed here for solution to the so formulated general second problem of portfolio selection under risk, differs from the so far known methods of global nonlinear optimization.

It reveals additional possibilities, related to the design and building of the new kind of informational services in Internet – portfolio selection under risk.

2. The models of the strictly convex, convex and non-convex problems of portfolio selection under risk

To input the goal function of the formulated problem, we will denote the portfolio variance as:

$$\sigma_p^2 = \frac{1}{2} x^T V x \,, \tag{1}$$

where

$x^T = [x_1, x_2, \ldots, x_n]$ is the vector of unknown allocation of the investment,

n is the number of securities in the portfolio,

x_i , i=1,2,…,n is the weight of the i-th security in the investment.

Negative values of x_i , i=1,2,…,n are acceptable, which corresponds to the so called short sales.

The negative value of x_i marks the sale of the i-th security and not its purchase and keeping.

If $x_i = 0$, the security is not included in the portfolio.

The value $x_i > 1$ marks availability of securities which amount is greater than the investor can buy.

This is possible only in the cases when the investor can obtain additional funds.

To fulfill the investment requirements of a long sales the solutions x need to be positively semi-determined, i.e. $x \geq 0$.

V is the symmetrical variance-covariance matrix of correlation among the various investment assets of dimensions n-by-n of the kind:

$$V = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} & \cdots & \sigma_{1n} \\ \sigma_{12} & \sigma_{22} & \sigma_{23} & \cdots & \sigma_{2n} \\ \sigma_{13} & \sigma_{23} & \sigma_{33} & \cdots & \sigma_{3n} \\ \vdots & \vdots & \vdots & & \vdots \\ \sigma_{1n} & \sigma_{2n} & \sigma_{3n} & \cdots & \sigma_{nn} \end{bmatrix}, \tag{2}$$

where σ_{ii}, i=1,…,n are variances and σ_{ij}, i=1,…,n; j=1,…,n are covariances.

In the strictly convex case the variance-covariance matrix V has only positive eigen values. In the convex case V may have positive as well as zero eigen values. In the non-convex case the symmetric variance-covariance matrix may have positive as well as negative eigen values.

The first new measure of quadratic risk (1) is introduced in [14], where the variance-covariance matrix is of the kind:

$$V_{new1} = \begin{bmatrix} \sigma_{11}/y_1 & \sigma_{12} & \sigma_{13} & \cdots & \sigma_{1n} \\ \sigma_{12} & \sigma_{22}/y_2 & \sigma_{23} & \cdots & \sigma_{2n} \\ \sigma_{13} & \sigma_{:23} & \sigma_{33}/y_3 & \cdots & \sigma_{3n} \\ \vdots & \vdots & \vdots & & \vdots \\ \sigma_{1n} & \sigma_{2n} & \sigma_{3n} & \cdots & \sigma_{nn}/y_n \end{bmatrix}. \tag{3}$$

where σ_{ii}/y_i , i=1,2,…,n are coefficients of variation of the first type.

The second new measure of quadratic risk (1) is introduced in [15,17], where the variance-covariance matrix is of the kind:

$$V_{new2} = \begin{bmatrix} \sqrt{\sigma_{11}}/y_1 & \sigma_{12} & \sigma_{13} & \cdots & \sigma_{1n} \\ \sigma_{12} & \sqrt{\sigma_{22}}/y_2 & \sigma_{23} & \cdots & \sigma_{2n} \\ \sigma_{13} & \sigma_{:23} & \sqrt{\sigma_{33}}/y_3 & \cdots & \sigma_{3n} \\ \vdots & \vdots & \vdots & & \vdots \\ \sigma_{1n} & \sigma_{2n} & \sigma_{3n} & \cdots & \sqrt{\sigma_{nn}}/y_n \end{bmatrix}. \tag{4}$$

where $\sqrt{\sigma_{ii}}/y_i$, i=1,2,…,n are coefficients of variation of the second type.

We will present the portfolio expected return as:

$$Y_p = x^T y, \tag{5}$$

where $y^T = [y_1, y_2, ..., y_n]$,

y_i is the expected return of each asset i=1,2,…,n.

As the goal of the portfolio management we set the maximization of the measure of William Sharpe [8,9], which is of the kind:

$$S_p = \frac{Y_p}{\sigma_p}, \tag{6}$$

where σ_p is the standard deviation of the portfolio as a measure of risk.

We will examine a task, which solves the structure of the optimal algebraic portfolio of securities from the viewpoint of the investor goals.

Task. To build a structure of an algebraic portfolio of securities where the expected return is calculated at a fixed acceptable minimal risk.

The formal description of the task is:

$$x^T y \geq Y_0 \tag{7}$$

where Y_0 is the minimum acceptable expected return of the portfolio.

We will assume that:

$$Y_0 = y_{min} \tag{8}$$

under the linear and nonlinear constraints of the kind:

$$y_{min} \leq Y_p \leq y_{max} \tag{9}$$

$$\sigma^2_{p\min} = \frac{1}{2}\left(x_1 V_{11} x_1 - \bar{x}^T Q_2^{-1} Q_1 V_{11} x_1 + x_1 V_{12} \bar{x} - \bar{x}^T Q_2^{-1} Q_1 V_{12} \bar{x}\right), \tag{10}$$

where the variance-covariance matrix V is presented in a block form of the kind:

$$V = \begin{bmatrix} V_{11} & V_{12} \\ V_{12}^T & V_{22} \end{bmatrix}, \tag{11}$$

where V_{11} is a scalar, V_{12} is a vector row, V_{22} is a square (n-1)x(n-1) matrix.

The matrices Q1 and Q2 are presented below.

The vector x is decomposed as follows:

$$x_1, \qquad \bar{x}^T = \left[x_2, \quad x_3, \quad \cdots, \quad x_n\right] \tag{12}$$

The constraints (9) can be violated in case the investor uses additional borrowed funds.

$$e^T x = 1, \tag{13}$$

where $e^T = [1, 1, ..., 1]$.

David Luenberger notes in [4], that when short sales are possible, most or all the optimal x_i, i=1, 2,… have nonzero values (positive and negative), so that nearly all assets are used. Vice versa, when short sales are not possible, typically a lot of weights are equal to zero.

3. Computer-mathematical PSM4A2-algorithm for the solution of the general second problem of portfolio selection under risk

Step 1. Forming the input arrays of algorithm data:

We will assume that the investor has evaluated in a certain way the expected return of each security, i.e. he has built the expected return vector y and has estimated the variance-covariance matrix V.

The values of the vector y and the square matrix V are calculated using statistical and historic data of the assets return over a certain period of time.

We will assume that the investor projects to obtain expected return $Y_p \geq Y_0$ at the following acceptable minimal risk:

$$\sigma^2_{p\min} = \frac{1}{2}\left(x_1 V_{11} x_1 - \bar{x}^T Q_2^{-1} Q_1 V_{11} x_1 + x_1 V_{12} \bar{x} - \bar{x}^T Q_2^{-1} Q_1 V_{12} \bar{x}\right)$$

Step 2. Determining the eigen values χ_i and eigen vectors-rows q_i of the variance-covariance matrix V on the base of correlations of the type:

$$q_i V = \chi_i q_{i,} \quad i = 1,2,...n$$
$$q_i = [q_{i1} \quad q_{i2} \quad q_{i3} \cdots q_{in}], \quad i = 1,2,...n$$

The eigen vectors need to be normalized so that their length is equal to one.

Step 3. The following scalars, vectors and matrices are formed:

V_{11}, V_{12},

$$Q = \begin{bmatrix} q_{11} & q_{12} & q_{13} & \cdots & q_{1n} \\ q_{21} & q_{22} & q_{23} & \cdots & q_{2n} \\ q_{31} & q_{32} & q_{33} & \cdots & q_{3n} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ q_{n-1,1} & q_{n-1,2} & q_{n-1,3} & \cdots & q_{n-1,n} \end{bmatrix},$$

where Q is a rectangular (n-1)-by-n matrix.

$$Q = [Q_1 \quad Q_2],$$

where Q_1 and Q_2 are of the kind:

$$Q_1^T = [q_{11}, \quad q_{21}, \quad q_{31}, \quad \cdots \quad q_{n-1,1}],$$

$$Q_2 = \begin{bmatrix} q_{12} & q_{13} & \cdots & q_{1n} \\ q_{22} & q_{23} & \cdots & q_{2n} \\ q_{32} & q_{33} & \cdots & q_{3n} \\ \cdots & \cdots & \cdots & \cdots \\ q_{n-1,2} & q_{n-1,3} & \cdots & q_{n-1,n} \end{bmatrix},$$

where Q_2 is a square (n-1)-by-(n-1) matrix,

The eigen vector-rows, (n-1) in number, forming the Q matrix, correspond to all eigen values $\chi_i, i = 1,2,...n-1$, except the least one $\chi_{\min}$.

Step 4. The following system of linear algebraic equations of the kind is formed:

$$A_2 x = d_2,$$

Where $A_2 = \begin{bmatrix} Q \\ \cdots \\ e^T \end{bmatrix}$,

Where e is n-dimensional single vector:

$$e^T = [1, \quad 1, \quad \cdots, \quad 1]$$

$$d_2^T = [O^T, \quad 1],$$

where O is (n-1)-dimensional zero vector:

$$O^T = [0, \quad 0, \quad \cdots, \quad 0].$$

Step 5. The above system is solved in three ways:

– by means of the algebraic expression of the following kind in the cases of long and short sales:

$$x_1 = A_2^{-1} d_2.$$

The relative error of this solution of the system of algebraic equations is calculated by means of an expression of the kind:

$$\left\| d_2 - A_2 (A_2^{-1} d_2) \right\|_2 / \left\| d_2 \right\|_2$$

– by Matlab left division solver for the case of long or short sales:

$$x_2 = A_2 \setminus d_2$$

The relative error is of the kind:

$$\left\| d_2 - A_2 (A_2 \setminus d_2) \right\|_2 / \left\| d_2 \right\|_2$$

– and by means of the Matlab-function lsqnonneg(A_2,d_2), which is applicable only for the case of long sales:

$$x_3 = lsqnonneg(A_2, d_2)$$

The relative error is of the kind:

$$\left\| d_2 - A_2 lsqnonneg\ (A_2, d_2) \right\|_2 / \left\| d_2 \right\|_2$$

The first two approaches allow the interpretation of the algorithms as information services in Internet.

Step 6.The following scalar and subvector are formed:

$$x_1, \qquad \bar{x}^T = [x_2, \quad x_3, \quad \cdots, \quad x_n].$$

Step 7. The minimal variance $\sigma^2_{p\min}$, the minimal standard deviation $\sigma_{p\min}$,the expected return Y_p and Sharpe measure S_p are estimated, using the formulas, given above.

The synthesis of the proposed algorithm is carried out with the help of the direct and inverse transformations, published and applied in [3].

References

[1]. Гатев Г., Изследване на операции, София,2003.

[2]. Дочев Д., Р. Николаев, Теория на риска, Варна, 2007.

[3]. Сотиров Л.Н., Синтез иерархических оптимальных структур стационарных динамических систем, Proceedings of The 4-th International Conference on Control and Computer Science, , Bucharest, 1981, p.44-49.

[4]. Luenberger D.G., Investment Science, Oxford university press,1998.

[5]. Markowitz H., Portfolio selection, Journal of Finance, 7th edition,1952, p.77-91.

[6]. Markowitz H., The optimization of a quadratic function subject to linear constraints. In Naval Research Logistics, Quarterly 3,1956, p.111-133.

[7]. Markowitz H., Portfolio Selection- Efficient diversification of investments, Second edition, Blackwell,1991.

[8]. Sharpe, W.F., G.J. Alexander, J.V. Bailey, Investments, Fifth edition, Prentice Hall International, Inc., 1999.

[9]. Sharpe, W.F., Portfolio theory and capital markets, McGraw-Hill, 2000.
[10]. Sotirova, S., L. Sotirov, Computer-mathematical modeling of one class investment processes, based on optimal singular adaptive M3A1N computer observer- Part 1, Part 2, International Conference on CompSysTech'07, Proceedings, Rousse, 2007, p. IIIB.6-1 – IIIB.6-6, p. IIIB.7-1 – IIIB.7-5.
[11]. Sotirova, S.L., L.N. Sotirov, Discrete modeling of one class financial-economic processes, based on optimal singular adaptive M3A1N computer observer- Part 1, Part 2, International Conference on Automatics and Informatics' 07, Proceedings, , Sofia, 2007, Vol. 1, p.IV-13 – IV-16, Vol. 1, p.IV-17 – IV-20.
[12]. Sotirov, L., S. Sotirova, One analytical solution of the non-convex and the convex first problem of Markowitz for portfolio selection under risk, International Scientific Conference "Informatics in the Scientific Knowledge", Proceedings, Varna Free University "Chernorizets Hrabar", Institute of Mathematics and Informatics of The Bulgarian Academy of Sciences, Varna, 2008, p.33-39.
[13]. Sotirov, L.N., S.L. Sotirova, Computer-mathematical approach to solving of one non-convex and one convex problem of portfolio selection under risk, International Conference on Automatics and Informatics' 08, Proceedings, , Sofia, 2008., p.XIII-5-XIII-8.
[14]. Sotirova, S.L., An analytical solution of one problem for portfolio selection under risk, Computer Science and Technologies, Varna, 2008, Number 1/2008, p.57-63.
[15]. Sotirov L.N.,S.L.Sotirova, One analytical solution to general second problem of Markowitz for portfolio selection under risk, , International Conference on Automatics and Informatics' 09, Proceedings, Sofia, 2009, p.V-17 – V-20.
[16]. Sotirova S., L. Sotirov, About singularity of two approaches to solving problems of portfolio selection under risk, International Conference on CompSysTech'09, Proceedings, Rousse, 2009, p. IIIB.19-1 – IIIB.19-9.
[17]. Sotirova S., L. Sotirov, A new measure of risk and its application in non-convex problems of portfolio selection, International Conference on Information Technologies in Business Management, University of Economics- Proceedings, Varna, 2010, p. 396-404.

For contacts:
Professor Lyubomir Nikolaev Sotirov, DSc, PhD,
Department of Computer Science and Engineering, Technical University of Varna,
Phone: +359 52 710 411, E-mail: l_n_sotirov@abv.bg

Stela Lyubomirova Sotirova-Nikolova, Master of Corporate Finance,
Bachelor of International Business, Bachelor of Computer Systems and Technologies,
Phone: +359 898 253 613, E- mail: stellasot@yahoo.co.uk

MSc., PhD Student,Samuil Vladimirov Nikolov,
Department of Computer Science and Engineering, Technical University of Varna,
Phone: +359 52 301 067, E-mail: samnikolov@yahoo.com

Assoc. Prof. Vladimir Nikolov Nikolov, PhD,
Department of Computer Science and Engineering, Technical University of Varna,
Phone: +359 52 309 288, E-mail: vnnikolov_bg@yahoo.com

Assoc. Prof. Trifon Ivanov Ruskov, PhD
Department of Computer Science and Engineering, Technical University of Varna,
Phone: +359 52 383 424, E-mail: ruskov@tu-varna.bg

MSc., PhD Student, Ivan Trifonov Ruskov,
Department of Computer Science and Engineering, Technical University of Varna,
Phone: +359 52 751 184, E-mail: i.ruskov@gmail.com

COMPUTER-MATHEMATICAL ALGORITHM FOR THE SOLUTION THE GENERAL SECOND PROBLEM OF PORTFOLIO SELECTION UNDER RISK-Part 2-APPLICATIONS

Lyubomir N. Sotirov, Stela L. Sotirova-Nikolova, Samuil V. Nikolov, Vladimir N. Nikolov, Trifon I. Ruskov, Ivan T. Ruskov

Abstract. The terms of strictly convex, convex and non-convex problems of portfolio selection under risk and new measures of risk are introduced. Algorithms for direct and iterative solving of the general second problem of portfolio selection under risk are proposed. The synthesized analytical solutions allows the interpretation of the investigated optimizational models of the expected return at fixed acceptable minimal risk as an information service in Internet.
Keywords. portfolio selection, strictly convex, convex and non-convex problems, expected return, risk, linear algebraic systems equations, information service in Internet.

Компютърно-математически алгоритъм за решаване на обща втора задача за избор на портфейл в условия на риск-част 2 -приложения

Любомир Н. Сотиров, Стела Л. Сотирова-Николова, Самуил В. Николов, Владимир Н. Николов, Трифон И. Русков, Иван Т. Русков

Резюме: Въвеждат се понятията строго изпъкнали, изпъкнали и неизпъкнали проблеми за избор на портфейл в условия на риск, както и нови начини за измерване на риска. Предлагат се алгоритми за директно и итеративно решаване на общата втора задача за избор на портфейл в условия на риск. Синтезираните аналитични решения позволяват интерпретирането на изследваните оптимизационни модели на очакваната възвръщаемост при определен приемлив минимален риск като информационна услуга в интернет.
Ключови думи: избор на портфейл, строго изпъкнали, изпъкнали и неизпъкнали проблеми, очаквана възвръщаемост, риск, системи линейни алгебрични уравнения, информационни услуги в интернет

4. A practical example with the classical quadratic measure of risk

Three different stocks are available. We will present their characteristics as follows:

$$V = \begin{bmatrix} 0.0050 & -0.0100 & 0.0040 \\ -0.0100 & 0.0400 & -0.0020 \\ 0.0040 & -0.0020 & 0.0230 \end{bmatrix}, \qquad V_{11} = 0.0050, \quad V_{12} = \begin{bmatrix} -0.0100 & 0.0040 \end{bmatrix},$$

$$y^T = \begin{bmatrix} 0.1, & 0.2, & 0.15 \end{bmatrix}, \qquad e^T = \begin{bmatrix} 1, & 1, & 1 \end{bmatrix}, \qquad O^T = \begin{bmatrix} 0, & 0 \end{bmatrix},$$

The eigen values of the matrix V are respectively equal to:

$$eig(V) = \begin{bmatrix} 0.0431 \\ 0.0231 \\ 0.0018 \end{bmatrix}.$$

The corresponding determinant is:

$$\det(V) = 1.8 * 10^{-6}$$

The corresponding conditioning number is :

$$\text{cond}(V)=23.8304,$$

We will assume that the investor wants to obtain expected return $Y_p \geq 0.1$ at the fixed minimal risk, given by the formula (10) in [18].

We can form the following data arrays for the above synthesized algorithm:

$$Q=\begin{bmatrix} -0.2656 & 0.9527 & -0.1477 \\ 0.1146 & 0.1833 & 0.9764 \end{bmatrix},\ Q_1=\begin{bmatrix} -0.2656 \\ 0.1146 \end{bmatrix},\ Q_2=\begin{bmatrix} 0.9527 & -0.1477 \\ 0.1833 & 0.9764 \end{bmatrix},$$

$$A_2=\begin{bmatrix} -0.2656 & 0.9527 & -0.1477 \\ 0.1146 & 0.1833 & 0.9764 \\ 1 & 1 & 1 \end{bmatrix},$$

$d_2^t=[0,0,1]$. $\det(A_2)=1.0418$, $\text{cond}(A_2)=3.5585$.

Then we can write the following about the solution of the second problem of portfolio selection under risk, obtained in Matlab environment by the first two ways, which interpret the direct solutions, which can be interpreted as information service in Internet:

$$x_1=x_2=\begin{bmatrix} 0.9189 \\ 0.2327 \\ -0.1515 \end{bmatrix}.$$

This is the case of short sales,

where

$$x_{11}=0.9189, \qquad \bar{x}_1^T=[0.2327,-0.1515].$$

The number of floating point operations when the left division solver is used, is:

flops=100.

The number of floating point operations when the respective algebraic expression is used for obtaing the same result, is:

flops=135.

Besides, the solution of the system of algebraic equations is obtained with error, equal to zero:

$\Delta_1=\Delta_2=0.0$.

The minimal acceptable risk as minimal variance is equal to:

$$\sigma^2{}_{p1\min}=8.3308.10^{-4}.$$

The values of the minimal standard deviation as a measure of risk, the expected return and the Sharpe measure are:

$$\sigma_{p1\min}=0.0289,$$
$$Y_{p1}=0.1157,$$
$$S_{p1}=4.0087.$$

We can write the following about the solution of the second problem of portfolio selection under risk, obtained in Matlab environment by means of the Matlab-function lsqnonneg(A_2,d_2), which interprets iterative solutions, which are not object of interpretation as information service in Internet:

$$x_3=\begin{bmatrix} 0.7750 \\ 0.2085 \\ 0 \end{bmatrix},$$

This is the case of long sales, where:

$$x_{31}=0.7750, \qquad \bar{x}_3^T=[0.2085,\quad 0].$$

The number of floating point operations is:
flops=1280.
The solution of the system of algebraic equations is obtained with error, equal to:
Δ_3=0.1283.
The minimal acceptable risk as minimal variance is equal to $\sigma^2{}_{p3\min} = 7.4088.10^{-4}$. The values of the minimal standard deviation as a measure of risk, the expected return and the Sharpe measure are:

$$\sigma_{p3} = 0.0272$$
$$Y_{p3} = 0.1192$$
$$S_{p3} = 4.3793.$$

5. A practical example with the first new quadratic measure of risk

We will present the first new variance-covariance matrix as follows:

$$V_{new1} = \begin{bmatrix} 0.0050/0.1 & -0.0100 & 0.0040 \\ -0.0100 & 0.0400/0.2 & -0.0020 \\ 0.0040 & -0.0020 & 0.0230/0.15 \end{bmatrix}, \; V_{new1} = \begin{bmatrix} 0.0500 & -0.0100 & 0.0040 \\ -0.0100 & 0.2000 & -0.0020 \\ 0.0040 & -0.0020 & 0.1533 \end{bmatrix}, \qquad V_{new11} = 0.0500,$$
$$V_{new12} = \begin{bmatrix} -0.0100 & 0.0040 \end{bmatrix}, \quad y^T = \begin{bmatrix} 0.1, & 0.2, & 0.15 \end{bmatrix}, \qquad e^T = \begin{bmatrix} 1, & 1, & 1 \end{bmatrix}, \qquad O^T = \begin{bmatrix} 0, & 0 \end{bmatrix},$$

The eigen values of the matrix V_{new1} are respectively equal to:

$$eig(V_{new1}) = \begin{bmatrix} 0.2008 \\ 0.1534 \\ 0.0492 \end{bmatrix}.$$

The corresponding determinant is:
det(V_{new1})=0.0015.
The corresponding conditioning number is :
cond(V_{new1})=4.0813.
We will assume that the investor wants to obtain expected return $Y_p \geq 0.1$ at the fixed minimal risk, given by the formula (10) in [18].
We can form the following data arrays for the above synthesized algorithm:

$$Q = \begin{bmatrix} -0.0674 & 0.9966 & -0.0477 \\ 0.0338 & 0.0501 & 0.9982 \end{bmatrix}, \; Q_1 = \begin{bmatrix} -0.0674 \\ 0.0338 \end{bmatrix}, Q_2 = \begin{bmatrix} 0.9966 & -0.0477 \\ 0.0501 & 0.9982 \end{bmatrix},$$

$$A_2 = \begin{bmatrix} -0.0674 & 0.9966 & -0.0477 \\ 0.0338 & 0.0501 & 0.9982 \\ 1 & 1 & 1 \end{bmatrix},$$

$d_2^t = [0,0,1]$. det(A_2)=1.0258, cond(A_2)=3.6235.

Then we can write the following about the solution of the second problem of portfolio selection under risk, obtained in Matlab environment by the first two ways, which interpret the direct solutions, which can be interpreted as information service in Internet:

$$x_1 = x_2 = \begin{bmatrix} 0.9721 \\ 0.0640 \\ -0.0361 \end{bmatrix}.$$

This is the case of short sales,where:

$$x_{11} = 0.9721, \qquad \bar{x}_1^T = [0.0640, -0.0361].$$

The number of floating point operations when the left division solver is used, is:

flops=100.

The number of floating point operations when the respective algebraic expression is used for obtaing the same result, is:

flops=135.

Besides, the solution of the system of algebraic equations is obtained with error, equal to zero: $\Delta_1= \Delta_2=0.0$.

The minimal acceptable risk as minimal variance is equal to:

$$\sigma^2{}_{p1\min} = 0.0234 \ .$$

The values of the minimal standard deviation as a measure of risk, the expected return and the Sharpe measure are:

$$\sigma_{p1\min} = 0.1529,$$
$$Y_{p1} = 0.1046,$$
$$S_{p1} = 0.6841.$$

We can write the following about the solution of the second problem of portfolio selection under risk, obtained in Matlab environment by means of the Matlab-function lsqnonneg(A_2,d_2), which interprets iterative solutions, which are not object of interpretation as information service in Internet:

$$x_3 = \begin{bmatrix} 0.9360 \\ 0.0628 \\ 0 \end{bmatrix},$$

This is the case of long sales,where:

$$x_{31} = 0.9360, \qquad \bar{x}_3^T = [0.0628, \quad 0].$$

The number of floating point operations is:

flops=1280.

The solution of the system of algebraic equations is obtained with error, equal to:

Δ_3=0.0348.

The minimal acceptable risk as minimal variance is equal to $\sigma^2{}_{p3\min} = 0.0217$. The values of the minimal standard deviation as a measure of risk, the expected return and the Sharpe measure are:

$$\sigma_{p3\min} = 0.1473$$
$$Y_{p3} = 0.1062$$
$$S_{p3} = 0.7206.$$

6. Conclusion

The solution to the second problem of portfolio selection under risk is investigated in the theory and practice of non-linear programming and is accompanied by a complicated iterative procedure, which hinders the interpretation of the second problem of Markowitz as an information service in Internet.

The general algebraic approach and method in the current paper allows the non-linear optimization problem of portfolio selection under risk to be solved by means of linear analytical algorithms at acceptable precision and velocity, which allows the interpretation of the general second problem of portfolio selection under risk as an information service in Internet.

It is important to note the effect from the usage of the first new measure of risk V_{new1}, which is the improvement of the precision of the iterative solution. In the common case, the precision of the iterative solution, as well as that of the analytical solutions improves. However, the Sharpe measure decreases, which can be compensated by the unique quality of the first new measure of risk to transform the non-convex quadratic forms of risk into strictly convex ones [14, 16, 17], which

extensively maximizes the application field of the portfolio optimization methods, worked out so far [1,2,12,13], as Internet information service, by significantly simplifying the proof of the sufficient conditions of Lagrange optimizational method.

References

[1]. Гатев Г., Изследване на операции, София,2003.
[2]. Дочев Д., Р. Николаев, Теория на риска, Варна, 2007.
[3]. Сотиров Л.Н., Синтез иерархических оптимальных структур стационарных динамических систем, Proceedings of The 4-th International Conference on Control and Computer Science, , Bucharest, 1981, p.44-49.
[4]. Luenberger D.G., Investment Science, Oxford university press,1998.
[5]. Markowitz H., Portfolio selection, Journal of Finance, 7th edition,1952, p.77-91.
[6]. Markowitz H., The optimization of a quadratic function subject to linear constraints. In Naval Research Logistics, Quarterly 3,1956, p.111-133.
[7]. Markowitz H., Portfolio Selection- Efficient diversification of investments, Second edition, Blackwell,1991.
[8]. Sharpe, W.F., G.J. Alexander, J.V. Bailey, Investments, Fifth edition, Prentice Hall International, Inc., 1999.
[9]. Sharpe, W.F., Portfolio theory and capital markets, McGraw-Hill, 2000.
[10]. Sotirova , S., L. Sotirov, Computer-mathematical modeling of one class investment processes, based on optimal singular adaptive M3A1N computer observer- Part 1, Part 2, International Conference on CompSysTech'07, Proceedings, Rousse, 2007, p. IIIB.6-1 – IIIB.6-6, p. IIIB.7-1 – IIIB.7-5.
[11]. Sotirova , S.L., L.N. Sotirov, Discrete modeling of one class financial-economic processes, based on optimal singular adaptive M3A1N computer observer- Part 1, Part 2, International Conference on Automatics and Informatics' 07, Proceedings, , Sofia, 2007, Vol. 1, p.IV-13 – IV-16, Vol. 1, p.IV-17 – IV-20.
[12]. Sotirov, L., S. Sotirova, One analytical solution of the non-convex and the convex first problem of Markowitz for portfolio selection under risk, International Scientific Conference "Informatics in the Scientific Knowledge", Proceedings, Varna Free University "Chernorizets Hrabar", Institute of Mathematics and Informatics of The Bulgarian Academy of Sciences, Varna, 2008, p.33-39.
[13]. Sotirov, L.N., S.L. Sotirova, Computer-mathematical approach to solving of one non-convex and one convex problem of portfolio selection under risk, International Conference on Automatics and Informatics' 08, Proceedings, , Sofia, 2008., p.XIII-5-XIII-8.
[14]. Sotirova, S.L., An analytical solution of one problem for portfolio selection under risk, Computer Science and Technologies, Varna, 2008, Number 1/2008, p.57-63.
[15]. Sotirov L.N.,S.L.Sotirova, One analytical solution to general second problem of Markowitz for portfolio selection under risk, , International Conference on Automatics and Informatics' 09, Proceedings, Sofia, 2009, p.V-17 – V-20.
[16]. Sotirova S., L. Sotirov, About singularity of two approaches to solving problems of portfolio selection under risk, International Conference on CompSysTech'09, Proceedings, Rousse, 2009, p. IIIB.19-1 – IIIB.19-9.
[17]. Sotirova S., L. Sotirov, A new measure of risk and its application in non-convex problems of portfolio selection, International Conference on Information Technologies in Business Management, University of Economics- Proceedings, Varna, 2010, p. 396-404.
[18]. Sotirov L., S.Sotirova-Nikolova, S.Nikolov, et.al. Computer-mathematical algorithm for the solution of the general second problem of portfolio selection under risk – part 1 - theory (In the same proceedings).

ИЗПОЛЗВАНЕ НА РАЗПОЗНАВАНЕ НА КОМПОНЕНТИ ЗА ЕВОЛЮЦИОНЕН АНАЛИЗ НА СОФТУЕРА

Тодор П. Чолаков, Димитър Й. Биров

Резюме: В практиката често се налага доработване и промяна на вече съществуващ, но непознат за разработчиците софтуерен продукт. За постигане на качество на промените е необходимо познаване на архитектурата на продукта и по възможност етапите на развитието му. В настоящия документ представяме подход за анализ на еволюцията на продукт, за който имаме само програмния код за някаква последователност от версии.
Ключови думи: обработка на код, реенжинеринг,еволюция на софтуера.

Using component recognition for analysis of software recognition

Todor P. Cholakov, Dimitar Y. Birov

Abstract: In practice it is often necessary to develop and change an already existing software product. In order to achieve the necessary quality of the changes it is necessary to have knowledge of the product architecture and if possible its evolution. In this paper we represent an approach for analysis of software evolution of a product if we only have its code base for set of versions.
Keywords: code processing, reengineering, software evolution.

1. Увод

В практиката често се налага да се разработва или доработва софтуер, който разработчиците не познават и за който единственото, с което разполагат, е програмният код за последователност от версии на продукта. Преди да започнат сериозни промени по такъв продукт, е необходимо да се извлекат знания за неговата архитектура и по възможност за неговата еволюция.

Еволюцията на софтуера е много важна от гледна точка на това, че чрез нея може да се обясни защо даден метод или клас изглежда точно по този начин, а не по друг.

Задачата, която си поставяме в този текст, е да извлечем знания за еволюцията на софтуера чрез автоматизираното разпознаване на съставните му компоненти при различни версии на обработвания софтуер и различни параметри на разпознаващия алгоритъм.

Прадположението, от което тръгваме, е че с времето свързаността в рамките на даден компонент се увеличава от центъра към периферията му, а самият компонент се разраства. Софтуерът като цяло се усложнява с времето, а компонентите му имат склонност по-скоро към сливане, отколкото към разделяне. Обратният случай - когато компонент се разделя на две или когато два компонента стават по-слабо свързани, е индикатор за съзнателно и целенасочено преструктуриране на кода.

Наблюдавайки еволюционното развитие на даден модул, както и конкретното място на даден код (клас) в рамките на модула, програмистът може да си създаде достатъчно добра представа каква е била идеята на този модул и как се е променяла с времето, което би му дало необходимото разбиране за по-нататъшни промени по кода.

Останалата част от настоящия текст е организирана както следва:

В секция 2 е описан накратко алгоритъмът за разпознаване на компоненти, който използваме. В секция 3 е описан подходът, който предлагаме за анализ на еволюцията на софтуер, чрез разпознаване на компонентите му за различни версии. В секция 4 е описан

експеримент, основан на този подход, като се разглежда кодът на няколко верисии на Ant. В секция 5 са описани насоки за бъдеща работа по проблема.

2. Алгоритъм за разпознаване на компонентите на софтуерен продукт

Алгоритъмът, който използваме разчита, като вход на граф на извикванията с тежест на всяко ребро, указваща броя на извикванията от кода, отговарящ на първия връх, към кода, отговарящ на втория връх.

Като параметър на алгоритъма задаваме минималната степен на свързаност φ, при която можем да приемем, че два върха принадлежат към един и същи компонент.

Първата стъпка на нашия алгоритъм включва преобразуване на входящия граф в нов, като в новия граф върховете остават същите, но ребрата и техните тежести съответстват на степента на близост между върховете. Степента на близост е метрика, която сме разработили специално за целите на разпознаване на софтуерни компоненти и ще я разгледаме в повече подробности по-долу. Като резултат от преобразуването на графа може да се появят нови ребра между върховете, както и сеиозно да се измени тежестта на някои от вече съществуващите. Докато оригиналният граф е ориентиран и теглата на ребрата между два върха са различни в зависимост от посоката, то резулататният граф е неориентиран, като метриката за степен на свързаност дава еднакъв резултат и в двете посоки.

Втората стъпка в нашия алгоритъм е да подредим върховете на графа (от тук нататък ще говорим само за резултатния граф след трансформацията) по намаляващ ред на тяхната максимална степен на свързаност с някой друг връх.

Като трета стъпка обхождаме получения списък и за всеки връх изпълняваме следното:

1. Вземаме реброто, отговарящо на максималната свързаност с текущия връх и проверяваме дали другия връх вече е поставен в някой компонент.
2. Ако вторият връх е поставен в компонент и степента на свързаност между двата е по-голяма или равна на φ, поставяме текущия връх в същия компонент.
3. Ако вторият връх не е поставен в компонент, но степента на свързаност между двата е по-голяма от φ, поставяме текущия връх в нов компонент.

След като сме преминали през целия списък, всеки връх има асоцииран компонент или е останал изолиран. Следва да се отбележи, че тъй като мярката за свързаност е симетрична и сме сортирали върховете на графа по максималната им степен на свързаност, то :

1. Всеки елемент, чиято максимална степен на свързаност е по голяма от φ, ще попадне в някой компонент.
2. Ако някой връх A сме го сложили сам в компонент, по точка 3 по-горе, то другият връх от реброто с максимална свързаност ще се появи като някой от следващите в списъка и неговата максимална свързаност ще е същата, като на A, поради симетричността на мярката за свързаност.

В литературата съществуват алгоритми, които намаляват стойността на φ и обхождат изолираните върхове итеративно, докато голяма част от тях попаднат в някой компонент [1, 2, 3]. Това от една страна подобрява клъстеризацията, като намалява броя на потенциалните компоненти, но от друга размива границите на компонентите. За целите на нашия анализ сме преценили, че е по-добре кодът, който не е достатъчно свързан с никой компонент, да се разглежда като отделен.

Основна роля в горния алгоритъм играе мярката **f** за свързаност между два върха **x** и **y** на графа. Тази мярка дефинираме по следния начин:

$$f(x,y) = w(x,y) * w(y,x) + co1(x,y) + co2(x,y) + rf(x,y) \quad (1)$$

Където

- $w(x,y)$ е броят на директните връзки от x към y

- $co1(x,y) = \frac{|out(x) \cap out(y)|^2}{|out(x)|+|out(y)|}$, където out(x) е множеството от всички върхове реферирани от върха x. Тази функция добавя степен на свързаност между два върха, когато едновременно реферираните от тях върхове са относително голям дял от изобщо реферираните върхове.

- $co2(x,y) = \frac{|in(x) \cap in(y)|^2}{|in(x)|+|in(y)|}$, където in(x) е множеството от всички върхове които реферират върха x. Тази функция добавя степен на свързаност между два върха когато множеството от едновременно реферищите ги върхове е относително голям дял от реферирращите ги върхове.

- $rf(x,y) = \frac{\sum_{t \in (in(x) \cup x)} w(t,y)}{|in(x)+1|} + \frac{\sum_{t \in (in(y) \cup y)} w(t,x)}{|in(y)+1|}$

Тази функция ще добави свързаност между двата върха, ако средната свързаност от някой от върховете и околността му към другия е относително голяма.

Основното предимство на така дефинираната мярка за близост е, че тя взема предвид не само директната връзка между два върха, но и между техните околности. По този начин се дефинират връзки и между върхове, които иначе не са директно свързани, но се ползват по подобен начин. Това елиминира до голяма степен и необходимостта от итеративно добавяне на върхове към компонентите.

3. Анализ на еволюцията на софтуерен продукт чрез разпознаване на компонентите му в няколко различни версии

Нашият подход за анализ на еволюцията на софтуера включва следните стъпки, които прилагаме върху всяка от версиите на продукта с които разполагаме:

1. Извличане на компонентите на софтуерния продукт за конкретната версия.
2. Извличане на компонентен граф на извикванията. Във въпросния граф върховете са разпознатите от първата стъпка компоненти, а дъгите мужду два върха са с тежест общият брой на извиквания от елементите на първия компонент, към който и да е елемент във втория.

След като получим така дефинираните графи от тяхната промяна с течение на времето (версиите на продукта), можем да извлечем следните знания:

- Даден компонент може да е бил разделен на няколко компонента. Това може да се определи автоматично, ако голяма част от елементите в даден компонент в следваща версия се намерят в два или повече други компонента. Тъй като в общия случай степента на свързаност между елементите в даден компонент не намалява, а при така дефинираната мярка за близост, за да се отделят два елемента, които са били в един и същи комонент, трябва да са разделят и използваните и използващите ги елементи. Тоест разделянето на компонент на части изисква доста усилия.
- Няколко компонента могат да се обединят. Това е доста по-вероятният случай, тъй като с времето връзките между класовете се увеличават.
- Някои компоненти може да са премахнати. Това също е рядко срещан случай в нормалния процес на развитието на софтуера.

Следващата стъпка, която предприемаме, е да анализираме компонентния граф на извикванията и промяната му с времето:

- Някои компоненти може да се свързват по-силно в течение на времето. Това е индикация, че евентуално в бъдещ момент те ще се слеят.

- Някой компонент може да става все по-малко свързан с останалите компоненти. Това е индикатор, че този компонент е все по-изолиран и евентуално в следващите версии може да отпадне.
- Компоненти или група от компоннети, които са изолирани от всички останали, вероятно са мъртъв код или модул, който все още е в процес на разработка.
- Ако връзките между компонентите вървят в една посока, това е индикатор за добра йерархична структура на кода.

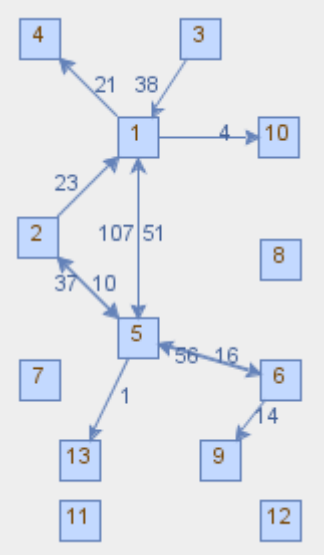
В следващата секция ще разгледаме реален пример с анализ върху няколко последователни версии на Ant.

4. Експериментален анализ на няколко версии на Ant, с помощта на горния подход

Изпълнихме алгоритъма за разпознаване на компоненти върху няколко версии на Ant със φ=0,5. Тази стойност на φ води до относително големи компоненти, така че можем по-лесно да проследим връзките между тях.

Във версия 1.1 получихме компонентен граф на извикванията със само 6 компонента, като повечето от класовете са в един компонент, а в останалите компоненти има по 1-2 класа. Това е индикатор, че първата версия на Ant се е състояла от силно свързано ядро и няколко класа, които не се използват. В следващи версии тези класове или ще сформират нови компоненти, или ще се присъединят към ядрото. Ако някои от тях останат изолирани в продължение на няколко версии, това означава, че са мъртъв код и следва да се махнат в следващата версия.

Във версия 1.3 получихме доста повече компоненти и компонентния граф изглежда както следва:

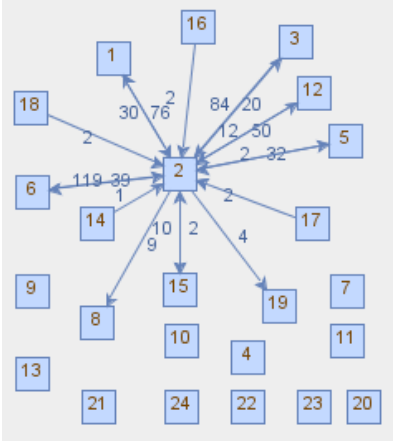


Фиг. 1. Компонентен гарф на извикванията за Ant 1.3

Един от компонентите, който разпознахме във версия 1.1, се е слял с основния компонент. Появили са се и няколко допълнителни компонента, които са свързани с основния. Както може да се види от графа, има няколко често използвани компонента – ядрото(1), визуалното ядро(5) и конзолата(2). Добавен е компонент ide(3), който също е силно свързан с ядрото. Забелязват се няколко компонента, които не са свързани с основния граф, които се състоят от по 1-2 класа. От графа на извикванията може да се направи извод, че архитектурата на софтуера в тази версия е централизирана около два основни компонента – ядрото (1) и визуалното ядро (5). Прави впечатление също ide компонента (3), който има връзки към ядрото, но към него няма връзки от никъде. Очевидно този компонент е замислен

като интерфейс за достъп към функциите на системата, тъй като в противен случай би бил излишен.

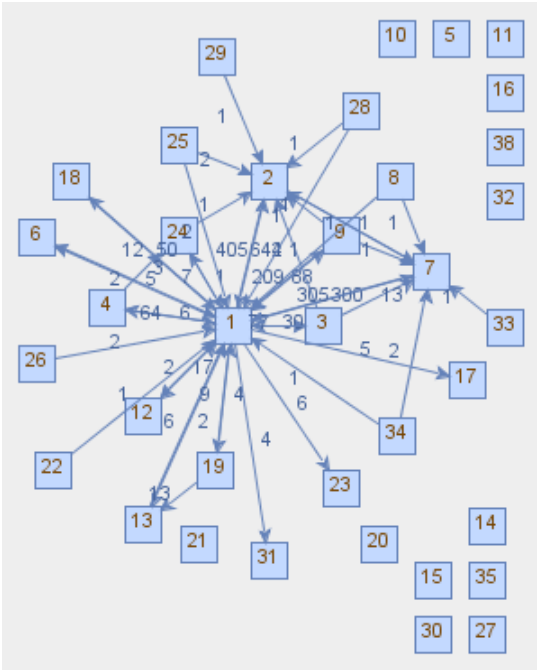
За версия 1.5 нещата изглеждат доста по различно:



Фиг. 2. Компонентен гарф на извикванията за Ant 1.5

Повечето от неизползваните компоненти от 1.3 са изчезнали, но на тяхно място са се появили множество нови неизползвани класове. Gui ядрото и конзолният компонент са изчезнали напълно. Архитектурата явно е доста по-централизирана, като всички компоненти ползват или са използвани единствено от ядрото. Отново се забелязват няколко компонента, които използват ядрото, но не са използвани от нищо, което и индикатор за потенциален мъртъв код или за интерфейси за външен достъп до системата.

Във версия 1.7 графа на извикванията изглежда много подобно:



Фиг. 3. Компонентен гарф на извикванията за Ant 1.7

Отново архитектурата е силно централизирана, въпреки че се появяват и директни връзки между компонентите, които са по скоро признак за ентропия, тъй като количеството им е пренебрежимо малко. Забелязва се разрастване на кода като обем и като брой компоненти, но при това не се променя основната архитектурна идея.

Като заключение можем да кажем, че с времето Ant е развил звездовидна архитектура и въпреки че от време на време се появява мъртъв код, той своевременно се отстранява.

5. Бъдеща работа

Като възможности за бъдещи разработки могат да се посочат две основни направления:

- Подобрение на алгоритъма за разпознаване на компоненти, чраз извличане на допълнителни знания за системата, включително автоматично именуване на компонентите.
- Автоматизиране на сравнението на компонентните графи и изводите основани на това сравнение.

6. Заключение

Графите на извикванията, комбинирани с алгоритъм за разпознаване на компонентите могат да дадат много добра представа за архитектурата на анализирания софтуерен продукт, както и за нейната промяна с течение на времето. Въз основа на предложения подход е възможно да се вземат информирани решения за промяна, преструктуриране и разработка на продукта.

Благодарности

Тази сатия е подпомогната и разработена по проект ДО 02-102/23.04.2009г. по ФНИ на МОМН.

Литература

[1]. Mitchel,B. On the Automatic Modularization of Software Systems Using the Bunch Tool. // IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, VOL. 32, NO. 3, MARCH 2006

[2]. Praditwong K., M. Hrman, X. Yao. Software Module Clustering as a Multi-Objective Search Problem // IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, VOL. 37, NO. XX, XXXXXXX 2011

[3]. Andreopoulous B., A. Aijun, V. Tzerpos, X. Wang. Multiple Layer Clustering of Large Software Systems. B: Proceedings of the Twelfth Working Conference on Reverse Engineeering, WCRE 2005, p.79-88

За контакти:
Тодор П. Чолаков, доц. д-р Димитър Й. Биров
катедра „Изчислителни системи”
Софийски университет „Св. Климент Охридски“
E-mail: todortk@abv.bg, birov@fmi.uni-sofia.bg

ЕДИН ПОДХОД ЗА СТЕГАНОГРАФСКА ЗАЩИТА НА ЦИФРОВИ ПРОДУКТИ

Емануил Ст. Стоянов, Божидар Ст. Стоянов

Резюме: Днес, повече от всякога, е лесно да се генерират цифрови данни. Това са т.н. „цифрово родени“ продукти - документи, изображения, аудио и видео записи, софтуер, електронни книги, цифрови модели, карти и т.н. Тяхното приложение се простира в различни области на нашето съвремие - бизнеса, науката, изкуството, образованието, цифровите комуникации, националната сигурност, разузнаването, военното дело и др. Заедно с това възникна необходимостта от тяхната надеждна защита, което е от първостепенно значение в съвременния информационен свят. Това наложи бързото развитие на две съвременни науки – криптография и стеганография. Този доклад разглежда един от съвременните варианти на стеганографията – цифровата стеганография, и предлага една идея за нейното приложение към всякакви цифрови продукти.
Ключови думи: Стеганография, скриващ метод, вплитане, цифров отпечатък.

An Approach to Steganographic Protection of Digital Products

Emanuil St. Stoyanov, Bozhidar St. Stoyanov

Abstract: Today, the generation of digital data is easier than it has ever been. These are the so-called digitally born products – documents, images, audio and video recordings, software, e-books, digital models, maps, etc. They are more and more widely employed nowadays in a variety of fields, such as business, science, art, education, digital communications, national security, intelligence service, military science, etc. Their global application has brought forth the need for their being reliably protected and has made it an issue of paramount importance in the modern world of information. This necessity has imposed the fast development of two contemporary sciences - cryptography and steganography. The present report treats one of the recent variations in the development of steganography - digital steganography, suggesting a universal idea for its application to all types of digital products.
Keywords: Steganography, Watermark, Fingerprint, Secret Image, Stego Object, Cover Image, Embeding.

1. Увод

Стеганографията е древно изкуство [1] и наука за предаване на скрити данни по такъв начин, че никой освен получателя да не разбере за съществуването им. Това е в противовес на криптографията, при която съществуването на данните не е скрито, а само тяхното съдържание. "Стеганография" е гръцка дума и означава [5] скрито писане (тайнопис). Корените и достигат до 400 г. пр. н. е., където древногръцкият историк Херодот споменава два примера.

През цялата история хората са крили информация по множество удивителни начини. Особен интерес за нас представляват съвременните стеганографски методи, които са напълно различни от древните. Днес, с развитието на компютърната техника, съвремените комуникации и Internet услугите, се обособи едно ново направление - цифровата стеганография. Тя изучава възможностите за скриване на информация в друга, анализирайки особеностите в цифровото представяне на данни, както и слабостите на човешките възприятия [6]. В този доклад е предложен подход, с помощта на който многократно се увеличават възможностите за приложение на цифровата стеганография – практически във всяка област от съвременния живот, където в електронен вид се съхранява и използва цифрово представена информация.

2. Изложение

Един от най-популярните съвременни методи за стеганографска защите е LSB (*Least Significant Bit*), основаващ се на използването на най-малко значимия бит [3] за представяне на скрити данни. Според други изследвания [4] съвсем успешно могат да се използват и последните два, без това да се забележи от човешкото око.

Методите за стеганографско скриване, които се използват в този доклад, са авторски и тяхното описание не е предмет на тази статия. Подробно са разгледани в [6]. Представляват поредица от трансформации за скриване на 8 битово изображение в 16 битово (респективно 16 битово в 32 битово). Базират се на особеностите в цифровото представяне на пикселите при преминаване от 8 битови RGB цветни канали в 16 битови, като съществено използват и слабостите на човешките възприятия [6].

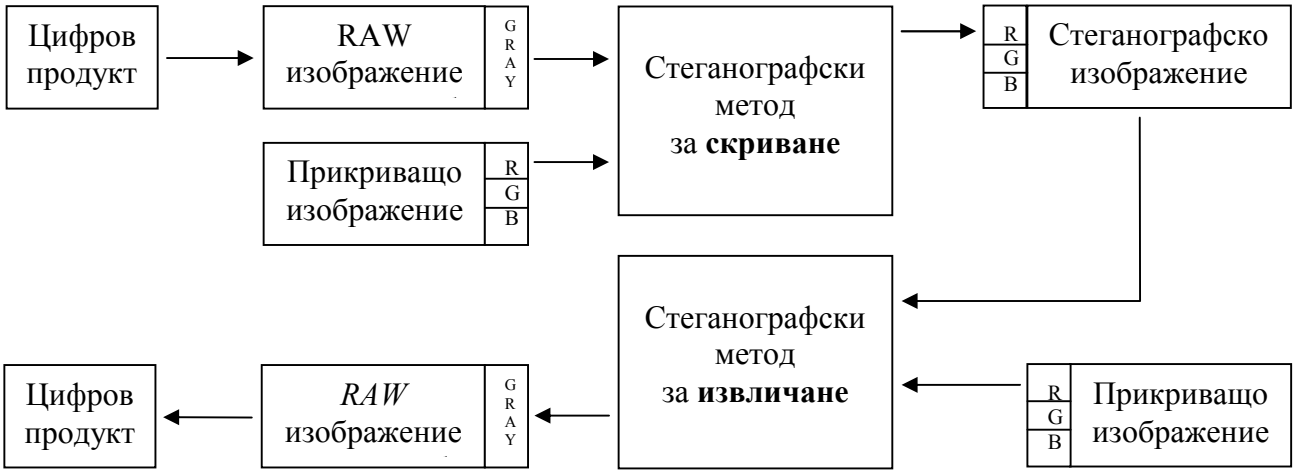
2.1. Основни понятия

Цифровата стеганография е относително млада област в информатиката. По тази причина все още няма разработена единна терминология и се налага предварително да бъдат формулирани всички понятия, с които се работи.

Цифровата стеганография може да използва разнообразно множество носители на скрита информация като изображения, звукови или видео файлове, IP пакети, транспортни слоеве (като файловете MP3) или протоколи (като UDP) и др. В статия [6] подробно са разгледани онези понятия, отнасящи се преди всичко към стеганография, базирана на цифровите изображения.

2.2. Един подход за скриване на произволен цифров продукт в изображение

Цифровите продукти могат да се представят като единични файлове с размер K bytes. В случаите, когато са многокомпонентни (например – софтуерните пакети), е възможно да се сведат до единствен файл чрез архивиране. Тъй като методите, използвани в този доклад, работят изцяло с изображения, ще сведем нещата до тях.



Фиг. 1. Принципна схема за прилагане на стеганографски методи към цифрови продукти

В основата на предлагания подход стои идеята, че всички файлове, без значение от тяхното предназначение, структура или тип, се съхраняват в електронен вид като последователност от байтове. В случаите, когато техните размери отговарят на условието от формула (2), тогава те могат да бъдат интерпретирани като растерни изображения с размери

MxN пиксела, чрез промяна на типа им в RAW – формат, некомпресиран (8-bits, Grayscale). В контекста на цифровата стеганография тези изображения са подходящи да бъдат обект на скриване и затова могат да бъдат наречени **секретни** (Secret Images). Процесът на скриване изисква да е налично и още едно изображение, наречено **прикриващо** (Cover Image). Нека то има размери PxQ пиксела. За да може Secret Image успешно да се „вплете" в Cover Image, задължително трябва да е изпълнено ограничението(3). Резултатът от тази операция е изображение неразличимо от прикриващото, което е прието да се нарича **стеганографско изображение** или Stego Object.

$$K = MN \tag{2}$$

$$M \le P,\ N \le Q \tag{3}$$

По този начин скриването на какъв да е цифров продукт се свежда до метод за вплитане на изображение в изображение. При обратния процес се извлича еквивалентно копие на вплетеното RAW изображение. Чрез повторна смяна на типа му към този, който е бил преди процеса на скриване, се възстановява оригиналният вид на файла, неговите първоначални формат и структура, което осигурява правилното му интерпретиране и го прави точно копие на скрития цифров продукт (Secret Image). Фигура 1 онагледява описания по-горе подход на скриване и извличане на произволен цифров продукт с изображение.

Разглеждането на произволен файл като Grayscale изображение допуска той да бъде скрит само в един от RGB каналите на цветно изображение. Това на практика означава, че в едно пълноцветно изображение могат да се скрият до 3 различни файла, удовлетворяващи условията (2) и (3).

Съществуват много и различни изисквания към качествата на разработваните стеганографски методи, по-важни от които са:

- запазване структурата на носещия (прикриващия) файл;
- скриване факта на присъствието на друга информация;
- устойчивост по отношение опитите за премахване или повреждане на скритата информация;

Таблица 1. Възможности за прилагане на стеганографски методи върху различни видове цифрови продукти.

Secret / **Cover**	*Изображение*	*Звуков файл*	*Изпълнима програма*	*Текстов документ*	*Архивен файл*
Изображение (RAW)	+	+	**!**	+	**!**
Изображение (не RAW)	-	-	-	-	-
Звуков файл (WAV)	+	+	**!**	+	**!**
Звуков файл (не WAV)	-	-	-	-	-
Изпълнима програма	-	-	-	-	-
Текстов документ	-	-	-	-	-
Архивен файл	-	-	-	-	-
Легенда	+ Възможно е без нарушаване структурата на файла				
	! Възможно е без нарушаване структурата на файла при определени условия				
	- Невъзможно е без нарушаване структурата на файла				

В Таблица 1 са показани възможните комбинации за съчетаване на два цифрови източника, отчитайки споменатите изисквания. От нея се вижда, че за прикриващи (Cover) източници могат да се използват файлове, в чиято структура липсва служебна информация (header). Такива формати са: RAW (за изображения), WAV (за звук), TXT (за текстов документ). Освен това съдържанието на носещите файлове трябва да е тясно свързано с несъвършенствата в човешките възприятия, каквито има при възприемането на образи,

звуци, сетивност, обоняние. Въпреки че текстовите документи имат подходящ файлов формат (TXT), те не могат да се използват за носители на скрита информация. Това е така, защото не е възможно да се удовлетвори второто изискване - скриване факта за наличие на друга информация.

От казаното дотук могат да се направят следните заключения:

- Най-неподходящи за ностели (Cover) са всички цифрови източници притежаващи header или такива, в които информацията се представя с твърде ограничено множество от стойности. Пример за това са текстовите файлове, при които отделните символи се представят само със стойности от 0 до 255.
- Всеки цифров източник е подходящ за скриване (Secret), стига да са удовлетворени условията (2) и (3), позоволяващи му да се представи в едноканално RAW - изображение.

2.3. Приложение на стеганографията за нуждите на различни отрасли

Цифровата стеганография се роди като наука буквално в последните години. Тя включва в себе си следните направления [5]:

– вграждане на информация с цел нейното скрито предаване;
– вграждане на цифрови водни знаци (watermarking);
– вграждане на идентификационни номера (fingerprinting);
– вграждане на заглавия (captioning).

В съвременните комуникации с бурно развиващи се мултимедийните технологии остро се поставя въпросът за защита на авторските права и интелектуалната собственост върху цифрово представените продукти. Особено актуален става въпросът, когато говорим за цифровата индустрия. Към нея спадат например софтуерната, фотографската, звукозаписната и киноиндустрии. Освен това защитата на авторските права е приоритетна област и за много други отрасли, които са само потребители на цифрови продукти. Такива са автомобилостроенето, самолетостроенето, геодезията, архитектурата и строителството, научно изследователски, военни организации и др.

2.3.1. Защита на авторските права и интелектуалната собственост

Предимствата, които дават представянето и предаването на цифрови данни, могат да се окажат засенчени от лекотата, с която е възможно тяхното откradване или модифициране. За това се разработват различни методи за защита, имащи организационно технически характер. Едно от най-ефективните технически средства за защита на цифрова информация се явява вграждането в защитаемия обект на невидим надпис-етикет (label), който се нарича цифров воден знак (ЦВЗ). Терминът <<digital watermark>> е бил използван за първи път в работата [7]. За разлика от обикновените водни знаци ЦВЗ могат да бъдат не само видими, но и (като правило) невидими. Невидимите цифрови водни знаци се анализират от специални декодери, които излизат с решение за тяхната коректност (автентичност). ЦВЗ може да съдържа в себе си автентичен код, информация за собственика, както и управляваща информация [5]. Най-подходящи обекти за защита с ЦВЗ са неподвижни изображения, файлове с аудио и видео данни. В тази връзка всички цифрови продукти, които могат да се сведат до цифрови изображения, също могат да се защитят с невидим цифров воден знак, указващ техния автор.

Друга технология, която има много общо с тази на ЦВЗ, използва вграждане на идентификационен номер от производителя на цифровия продукт. Разликата се заключава в това, че при <<*fingerprinting*>> всяко защитено копие има вграден уникален идентификационен номер, от където произлиза и названието – буквално <<*пръстов отпечатък*>>. Тези номера позволяват на производителите да проследяват бъдещата съдба

на своите продукти - не се ли занимава някой от купувачите с незаконно копиране (тиражиране). Ако е така, то <<*пръстовият отпечатък*>> бързо ще покаже виновника.

Тези методи са приложими във всички отрасли, които се занимават със създаване и комерсиалното разпространение на готови цифрови продукти.

2.1.1. Приложение на стеганографията в секретните комуникации

Скрити комуникации се използват не само от военни, шпионски и разузнавателни организации, но и от различни държавни институции като Президентство, някои министерства (МВР, МВнР), външна дипломация и др. Не рядко се използват и от бизнесмени, ръководствата на държавни и частни фирми с оглед опазване на фирмената тайна, ноу-хау, изобретения и др.

Секретна комуникация може да се осъществява както чрез криптиране на предаваната информация така и чрез обезпечаване секретността на самия трафик. Перспективен способ за осигуряване секретността на комуникацията е скриването на самия факт, че се осъществява секретна комуникация. Тази идея е залегната в самата същност на стеганографията – да скрива посланието по начин, който не позволява да бъде видяно [2].

При криптографията е в сила обратния принцип - разбърква се посланието по начин, който не позволява да бъде разбрано. Основен недостатък при използване на криптографията е, че предаването на секретна информация е открито (видимо) за неоторизирани потребители и лесно могат да я прихванат, копират и атакуват.

Отчитайки предимствата и недостатъците на двата метода (криптиране и стеганография), добър подход е съчетаването им за постигане на максимална сигурност и защита.

3. Резултати

На базата на предложения в точка 2.2 подход за скриване на произволни цифрови продукти в изображения, и разработения авторски метод, описан в [6], бяха реализирани успешно следните експерименти, описани в Таблица 2. За всеки от проверените случаи бяха констатирани пълно скриване и 100% възстановяване на скритата информация.

Таблица 2. Проведени експерименти

Области на приложение	**Описание на експеримента**
Фотограметрия	Скриване на секретно изображение в цензурирано.
	Скриване на стереодвойка в единична аерофотоснимка.
Геодезия	Скриване на DEM (Digital Elevation Model) с висока резолюция в DEM с ниска резолюция.
	Скриване на информация в цифрови карти.
Звукообработка	Скриване на звуков файл в изображение.
	Скриване на звуков файл в друг звуков файл.
	Скриване на изображение в звуков файл.
	Скриване на текстов файл в звуков.
	Скриване на изпълнима програма в звуков файл.
Растерна обработка	Скриване на изображение в изображение.
	Скриване на звуков файл в изображение.
	Скриване на изпълнима програма в изображение.
	Скриване на текстов файл в изображение.

4. Заключение

Животът на съвременното информационно общество е немислим без развитието и постиженията на новите цифрови технологии. Големите предизвикателства са най-вече около въпросите за защита на информацията, авторските права и поверителността на комуникацията. Подходът, който е представен в този доклад, се базира на съвременните стеганографски методи и представлява една стъпка към разрешаването на тези предизвикателства.

Благодарности

Изследванията и дейностите в този доклад са финансирани по проект РД-08-256/14.03.2013 от параграф ”Научни изследвания” на Шуменски Университет.

Литература

[1]. Neil F. Johnson, Sushil Jajodia. “Exploring Steganography: Seeing the Unseen”. Computing Practices. //IEEE Press, February 1998, pp.26-34.
[2]. T. Aura, “Invisible Communication”. // EET 1995, technical report, Helsinki Univ. of Technology, Finland, Nov.1995; URL: http://deadlock.hut.fi/ste/ste_html.html.
[3]. W. Brown and B.J. Shepherd, “Graphics File Formats: Reference and Guide”. // Manning Publications, Greenwich, Conn., 1995.
[4]. W. Bender et al., “Techniques for Data Hiding”. // IBM Systems J., Vol. 35, Nos. 3 and 4, 1996, pp. 313-336.
[5]. Грибунин В.Г., Оков И.Н., Туринцев И.В. “Цифровая стеганография” - М., "СОЛОН-Пресс", 2002.
[6]. Стоянов Б., Стоянов Е., “Обратими растерни трансформации и тяхното приложение в стеганографията”. // Международна научна конференция РУ "Ангел Кънчев" и Съюз на учените в Русе, 31.10.2008-01.11.2008 - гр. Русе., том 47, серия 3.2, с. 84-91.
[7]. Osborne C., van Schyndel R., Tirkel A., “A Digital Watermark”. // IEEE In-tern. Conf. on Image Processing, 1994. P. 86-90.

За контакти:
гл. ас. Емануил Ст. Стоянов,
катедра „Компютърна Информатика”,
ШУ „Еп. Константин Преславски”,
E-mail: emanuil_stoyanov@yahoo.com

гл. ас. Божидар Ст. Стоянов,
катедра „Геодезия”,
ШУ „Еп. Константин Преславски”,
E-mail: bojidar_stoyanow@yahoo.com

МЕТОДИ ЗА ПОЛУЧАВАНЕ НА МЕДИЦИНСКИ ОБРАЗИ, СРАВНИТЕЛЕН АНАЛИЗ И КОНТЕКСТУАЛНИ ПРОБЛЕМИ В ПОЛУЧЕНИТЕ ИЗОБРАЖЕНИЯ

Гергана В. Маркова, Мирослав Н. Гълъбов, Маргарита К. Тодорова

Резюме: Процесът на получаване на образ при фиксирани условия, отделяне на региони, представляващи интерес, класифициране и даване на експертна оценка след анализ, е изключително труден процес в областта на медицинската диагностика. В настоящата разработка е направено проучване, като са разгледани факторите, повлияващи получаването на медицински изображения в различните модалности във физичен и информатичен аспект, същността и контекста на образите, както и смисъла на екстерналните изображения. Направени са изводи за необходимостта от доразвиване на методите при получаване и постобработка на заснетите статични обекти с Digital Single Lens Reflex (DSLR).
Ключови думи: Медицински изображения, Медицински информационни системи, сегментация

Methods of medical images obtaining, a comparative analysis and contextual problems in the received images

Gergana V. Markova, Miroslav N. Galabov, Margarita K. Todorova

Abstract: The process of image obtaining in fixed conditions, separation of regions of interest, classification and giving of an expert assessment after analysis is an extremely difficult process in the field of Medical diagnostics. In the present study a research has been made and the factors which influence the obtaining of medical images in the different modalities in a physical and informatics aspect, the essence and the context of the images as well as the meaning of the external images have been examined. Conclusions have been reached about the necessity of a further development of the methods in the obtaining and post-processing of the shot static objects with DSLR.
Keywords: Medical images, Medical Information Systems, segmentation

1. Увод

Получаването на изображения, които се нуждаят от експертен прецизен анализ, е сложен процес. Той обхваща планиране на дейността, което включва задачата, която трябва да се разреши, подходяща апаратура, допълнителни аксесоари и условия, оператор, определящ настройването и, обект и анализатор-експерт. Всеки един от тези фактори допринася за коректното решение на задачата. Формирането на образа в съзнанието на човека е еволюционно програмирано, въпреки индивидуалните му особености, свързани с възприятията му. *Като елементарно възприятие* се счита възприятието на цвета, тъй като се свежда до оценка на осветеността, цветовия тон, наситеността на тона. Визуалната система на човека е адаптирана към възприемането на електромагнитно излъчване в определен диапазон на видимата светлина. Тя осигурява повече от 90% сензорна информация на главния мозък. Този процес започва с проекцията на изображението върху ретината на окото и завършва с обработката от висшите кортикални области, които вземат системни решения за формиране на визуален образ. Обособени са две групи перцептивни операции: *оценка на разстоянието до обекта* и *посоката*, в която се намира. Комбинацията от тези данни формират знанието за пространственото положение на обекта и неговото движение. *Най–сложен се счита процесът* при възприемането на формата на обекта. След фиксирането на

детайлите се установяват и пространствените отношения между тях [1]. Въпреки уеднаквените в еволюционен план възприятия, на всеки индивид при разпознаване на дадени обекти, влияят група от фактори, които в една или друга степен могат да затруднят неговите възприятия. Този факт е пряко свързан със вземане на решения при разчитане на образи от медика-специалист. Всяко технологично подобрение на апаратната част и софтуера допринася за по-точната диагностика.

Медицинските изображения (МИ) принадлежат към различни „множества". От една страна те са част от Медицинските информационни системи (МИС), от друга техният контекст определя здравния статус на пациента и от трета – те са сложен обект за създаване, обработка, анализиране, съхраняване и архивиране.

2. Образите в интегрираните медицински информационни системи

Възникването и усъвършенстването на интегрираните МИС е предизвикано от необходимостта за съхраняване, предаване и архивиране на данни на пациенти. Те са свързани пряко с Е-здравеопазване, Кибер-медицина, мобилни устройства за здравни цели, Телемедицината. Модулите за образи в тази сложна система имат въведени стандарти, които разрешават не само съхраняването и обмена, но частичната постобработка на получените образи. Такива примери са PACS (Picture Archiving and Communication System), ренгенологично-информационни системи - RIS, средства за обработка на снимкови резултати от ядрена диагностика [27], системи за визуализация на медицинска информация, системи за реконструкция и получаване на 3D образи и телерадиологични изследвания. Един от разработчиците е IMAGE Information Systems [23]. Предлагат се различни модули, обезпечаващи: цифровизиране на аналогово видео, обработка на резултатите в определена модалност, компресиране на изображения за телерадиологията, управление на работните процеси и др. Например модулът iQ-STITCH свързва няколко радиографии в едно изображение.

Практиката показва, че все още липсват на всички нива уеднаквени стандарти и несъвместимите апаратни и софтуерни бази на крупни разработчици като General Electric, Toshiba, Philips, Siemens PICKER затрудняват интеграцията на медицинските устройства при трансфер и визуализиране на данни. Стандартът DICOM-3.0 [3], 2011г. (Digital Imaging and Communication in Medicine) е специфичен и се отличава от общоприетите формати BMP, TIFF, JPG, GIF, PNG. Този стандарт съхранява допълнителна информация за действителните размери, позиционирането, плътността, мащабността, режима на заснемане и др. параметри на обекта, както и персонални данни за самия пациент. Предимството на този формат е, че в един файл в "хедъра" му се съдържа информацията за пациента, вида на сканинга, размера на изображението, а в "тялото" - данните за 2D или 3D-образ, за разлика от предхождащия го стандарт Analyze format, при който данните са разделени в два файла - *.img *.hdr. Друго предимство е, че един файл DICOM може да съдържа компресирана информация със загуба или без загуба на информация.

3. Същност и контекст на образите в медицината

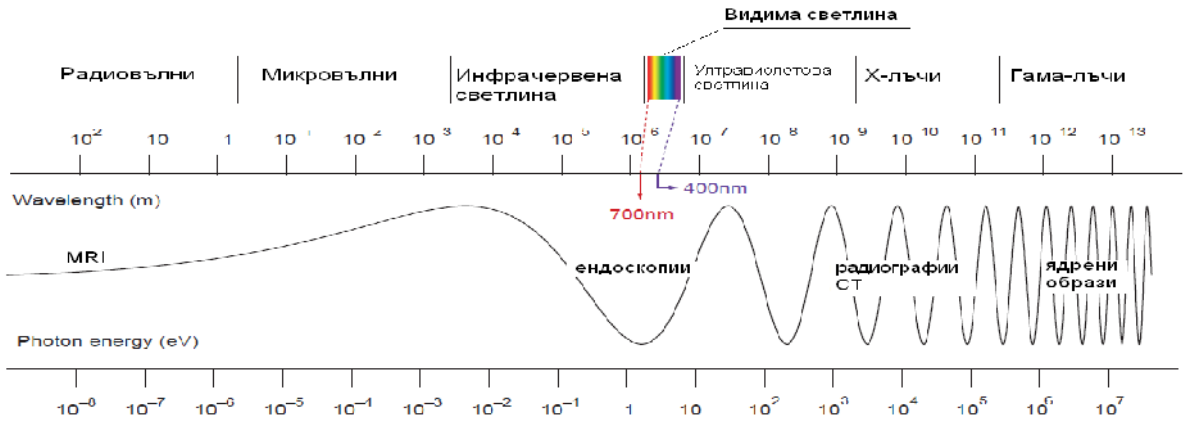
Към настоящия момент в медицината все по-често се разчита на изображения, за да се открие, проследи, сравни и анализира наличие на един или друг обект на интерес. В исторически план напредъкът в развитието на добиването на МИ се дължи на "технологичния натиск", оказан от развитие в други приоритетно финансирани области.

Могат да се дефинират няколко причини, които обуславят все по-голямата значимост на МИ:

1. Нарастващата сложност на биологичното познание, свързано с разбиране на уникалността на човешкото тяло и неговите статични и динамични свойства.
2. Напредъка в еволюционното развитие на изобразяващите технологии и задълбочаването им във все по-базисни функционални нива.
3. Ускорено развитие на компютърните технологии и мрежовия обмен, като: три- и четиримерни представяния; възможно наслагване на изображения от различни устройства; създаване на среди за виртуална реалност; транслиране на изображения на отдалечени обекти в реално време в mHealth и телемедицината.
4. Изображенията се налагат като ефективни средства с нарастващо значение за предаване на информация във визуално-ориентирани развити култури.
5. Все по-растящо количество информация за пациентите, част от което може да се помести в метаданните на изображения или във видимата им експозиция.

Получените образи на системата „Човешко тяло" са сложен обект от физична гледна точка с неговите характеристики [5] - прозрачност, непрозрачност, излъчвателна способност, отражателна способност, проводимост, намагнетизиране, както и промени в тези характеристики с течение на времето. За да се получи информация за основните свойства на обекта и образите в него, показващи една или повече от тези характеристики, се налага техният анализ.

На фиг. 1. е показан електромагнитният спектър и типовете модалности с техния периметър на ползване за получаване на образи. Към схемата може да се добави и *екстерналното заснемане* в сектора на видимата светлина, въпреки че в медицината не е прието за отделна модалност, но съществува, макар и не добре описано и развито.



Фиг. 1. Електромагнитният спектър и модалностите в неговото пространство [4]

4. Методи за генериране на медицински образи и проблеми при изграждане на образа.

Методите за генериране на МИ и тяхната дообработка могат да се разделят в две групи, в зависимост от *времевия фактор при тълкуването* - веднага ли се разчита от специалист или може да се приложи допълнителна софтуерна обработка преди анализа.

1. Използване на високотехнологични методи при първоначално генериране на образ, които максимално (според възможностите на апарата) намаляват шума, повишават контраста, и по този начин изясняват зоните на интерес в даден образ.
2. Тази категория включва горепосочените методи, като разширява избора от възможности и включва допълнителен специализиран софтуер за откриване на ръбове и сегментация на изследвания обект.

Рентгенови Х-лъчи, радиоактивността, магнитният резонанс, ултразвукът с развитието на инструментариума за визуализиране, предоставят днес едни от най-ефективните

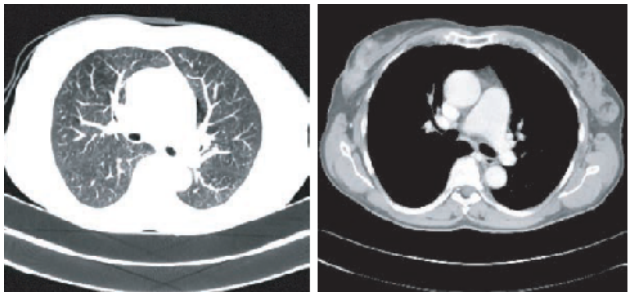
инструменти за диагностика и лечение. Съществуват: *Рентгенова компютърна томография* (X-ray CT), *Емисионна компютърна томография* (ECT), *Позитронна емисионна томография* (PET) - получаване на образ чрез използване на радиофармацевтик и позитронен разпад, *Однофотона емиссиона компютърна томография* (SPECT) при всеки акт на радиоразпад излъчва само един гама-квант.

При получаване на *рентгеновите изображения* вълните се абсорбират от плътни структури като кости, но преминават през по-малко плътните области. Според позицията на заснемане се наслагват пространствено обекти. При търсене на по-плътни структури, например при мамография, в тъкан винаги се наслагват образи по две причини – наличие на 3D структура, репродуцирана в 2D пространствен образ и допълнителното пресиране при този вид изследване. Повечето цифрови МИ днес използват 4096 степени на сивата скала [4, 9]. Проблем е, че малките разлики в яркостта не могат да се възприемат от класическия дисплей, както и възможностите на визуалния апарат на специалиста да разпознае тези степени.

Рентгеновите апарати, от които са получени рентгенографски снимки, имат следните проблеми в образа:

- пространствената разделителна способност е по-малка, поради ограниченията на разделителна способност на камерата;
- поради допълнителното конверсиране (лъчение → електрони → лъчение), шумът се увеличава;
- появяват се геометрични изкривявания, наречени ПИН-уплътнено изкривяване, по-специално към граничните ръбове на цялото изображение.

При *CT* чрез позиционираане на апаратите, се заснемат няколко проекции от различни ъгли (позиции) с цел изграждане на срезове в различни равнини, което не е възможно с обикновен рентген. На фиг. 2а,б е илюстрирана една зона на заснемане, но с различни обекти на интерес.



а) б)

Фиг. 2. а, б CT-образ на гръден кош в два различни режима: а) визуализират се белите дробове б) визуализират се меките тъкани [4]

Значението на *CT-методите* е свързано с няколко от следните функции:

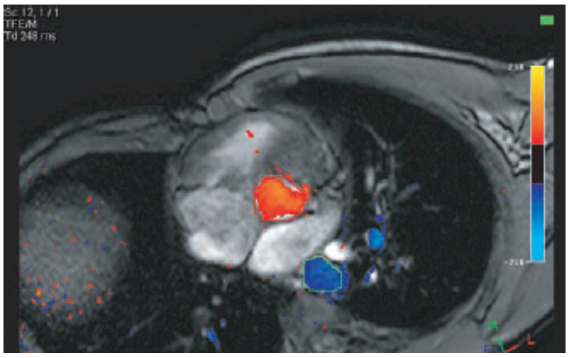
- ✓ предоставяне на срезов образ;
- ✓ наличие на висок контраст, превъзхождащ традиционната радиология;
- ✓ създаване на клинични образи, чрез междинен контрол на физичните опции и режимите на работа;
- ✓ визуализация на дигитални образи, които могат да се дообработят, за да се достигне търсениият детайл в информацията.

MRI-методите имат приложение за анализ чрез използване на морфометрия и автоматизирана сегментация на тъкан [11, 12]. Целта е сегментирането на отделни участъци, като се сече равнина отгоре надолу. В отделните срезове се различават обособени участъци, дефинирани в сивата скала, от чиито размер, положение, плътност и форма се съди правилност или специфика и патология на определени анатомични структури. В резултат на

използваните автоматизирани техники се разглеждат в някои случаи качествени, а в други – количествени параметри. Генерираните данни се компресират. Получените изображения могат да са с размерност 256 x 192 dpi при ъгъл на заснемане 45^0, както и 124 x 124 dpi с приложено интерполиране – 254 x 256 dpi [14]. Техниките за сегментиране и в МИ могат да бъдат разделени в класове:

- Ръчни, полуавтоматични и автоматични.
- Pixel-базирани (локални методи) и базирани на области (глобални методи).
- Ръчно очертаване, ниско ниво на сегментация (прагово, обединяване на области и др.), както и въз основа на модел (мултиспектрални, динамично програмиране, следване на контура и т.н.).
- Класически (прагово, край-базирани), статистически, размити множества, невронни мрежи.

Сегментирането при медицински образи обхваща всички познати методи и техники, като не съществува универсално приложим алгоритъм при прилагане на даден вид сегментация. На фиг. 3. е показан образ на аорта, като в червен цвят се вижда входящият кръвен поток, а в син (долу-център) – низходящият. Скалата вдясно служи за определител.



Фиг. 3. MRI-образ на аорта

FMRI предимно се използва за визуално представяне на моделирана динамична мозъчна активност, което е труден технически проблем. Неговата сравнително висока пространствена разделителна способност дава възможност за разработването на алгоритъм, който показва, наблюдаваните от пациента движещи се изображения [13] - фиг. 4. Този модел може да се използва за реконструкция на други видове динамични изображения (като сънища и спомени).

Фиг. 4. Приблизително се определя какви движещи се образи вижда пациентът

Последните достижения в областта на Облачните системи коренно изменят подходите за съхраняването и достъпа на данни, в това число и МИ. През 2011 година е стартирал проект за въвеждане на mHealth чрез използване на мобилни телефони, смартфони, комуникационни сателити, пациентски монитори в страни и области с много отдалечено или липсващо здравеопазване [28].

5. Фактори, определящи качествените характеристики на изображението, получено с различни устройства (модалности)

Както бе посочено, в образните медицински модалности се използват различни методи, основани на определен вид излъчване на сигнал/вълна, за да се достигне до визуално представяне на статичен образ или процес в динамика. Общото е, че независимо от начина на добиване и възможността за допълнителна дообработка, се стига до визуалното им представяне. В последният етап на анализ от специалист възниква задачата за субективно разпознаване на отделяне на обекти и техните граници.

Остротата в образа е един от важните фактори при формиране и разчитане на изображението. Факторите, допринасящи за наличието и, могат да се сведат до следните [5]:

1. характеристиките на системата, обработваща образите: точка на фокусиране и размера на детектора;
2. геометрията и характеристиките на сцената: формата на обекта, неговата позиция и движение;
3. условията за наблюдение.

Контрастът се определя като разликата в интензивността на съседни области в изображението. Това е амплитудата на Фурие-трансформация в изображението, като функция на пространствена честота. От друга страна той се изразява в разликите в яркостта и/или цвета, който прави един обект различим и се определя от [6]:

1. процеса на обработка на образа - източник на интензитет и ефективна абсорбция или чувствителност при улавяне от устройството;
2. характеристики на сцената - физични свойства, размерът и формата на обекта, използването на контрастни вещества;
3. условия - осветление и специализирано оборудване с дисплеи.

При апаратите, използващи Х-лъчи, спектърът на лъча повлиява контраста. От значение е плътността и дебелината на тъканните слоеве, които той среща по протежение на проекционната линия. Така наречената *мека радиация* (с голяма дължина на вълната) се използва при мамографии и там контрастът е по-висок, в сравнение с този при прилагане на твърда радиация. Друг фактор, който влияе на контраста, е ефективността на усвояването на детектора. При по-висока абсорбционна ефективност се получава по-висок контраст. При СТ-образа контрастът между един обект и неговия фон зависи предимно от съответните затихващи свойства, но също и от множество физични фактори като например спектъра на рентгеновата тръба и колко лъча са разпръснати.

Шумът в изображението е свързан с излъчването и откриване на допълнителни светлинни потоци и други видове електромагнитни вълни, които се определят като стохастични процеси. При прилагане на някои от техниките с увеличаване на шума се подобрява контрастът, но в тези случаи често са премахват малки детайли, а в други - премахването на шума намалява рязкостта на ръбовете и се губи информация за артефактите като цяло. Т.е. ако нивото на шум е високо в сравнение с интензитета на образа, тогава се губи важна информация. При *рентгенографиите* квантовият шум има статистически характер и е обикновено доминиращ шумов фактор. Процесът на фотонното откриване е всъщност Поасон-процес [7]. Стандартното отклонение на фотоновия шум е равно на корен квадратен от средния брой фотони. Отношението сигнал към шум (Signal-to-Noise Ratio - SNR) тогава е:

$$SNR = N/\sqrt{N} = \sqrt{N} \tag{1}$$

където N е средният брой събрани фотони. Когато N е много голямо, отношението сигнал към шум е също много голямо. Фотоновият шум става по-значим, когато броят събрани

фотони е малък. От тук и изводът: Това обяснява защо доза не може да бъде намалява без последствие за контраста. По този начин ще се намали SNR на неприемливо ниво.

В *CT-образите* се появяват три вида шум: квантов шум, електронен шум и шум получен в резултат от ограничения динамичен обхват на детектора. Факторите, повлияващи шумовете са следните [4]:

1. Общата експозиция. Повишаването на mA/s увеличава SNR. По този начин се намалява относителният квантов шум, но за сметка на дозата (облъчването) на пациента. При висококонтрастни изображения облъчването и високите нива на шума могат да бъдат толерирани чрез Реконструкционен алгоритъм.
2. Прилагане на Реконструкционен алгоритъм (РА) – двойка прилагани филтри и интерполационен метод оказват влияние върху шума в изображението. РА превръща измереният шум в структуриран шум в изображението.

Налага се изводът, че възможността за откриване на ниско-контрастна информация е много по-висока при CT, отколкото в радиографията. Основната причина е, че при втората в изображението се наслагват множество структури.

При *PET* има наличен фонов шум, но той позволява разграничаването на различните тъкани. Съществува и т.н. Speckle шум, който намалява качеството на изображението. Разработени подходи като Bayesshrink Wavelet Threshold [8] намаляват до минимум наличието му.

Артефактите и техният произход. Примери в медицински изображения са артефакти като метални жилки в CT-изображения и геометрични деформации в MR-изображения [10]. Нежеланите ефекти върху образа могат да се появят и в следствие на цифровата обработка на образа, като поява на погрешно изостряне на ръбовете. При рентгенографиите получените артефакти произлизат от драскотини в детектора, мъртви пиксели, непрочетени сканирани линии, нехомогенен рентгенов интензитет. CT-артефакти са: Stairstep артефакти – стъпаловидно накъсване на цвета по наклонени ръбове и повърхнини, както и при 3D изображения; Bearclaw артефакти – проявяват се като стъпаловидни светлинни конуси; артефактът „цъфтеж” е много често срещан и се проявява при образ на калцирана плака. Това се дължи на комбинация от засилване на лъча и ограничената пространствена разделителна способност [9]. Две от най-честите причини за наличие на артефакти в SPECT и PET са Поасонов шум и движението на пациента по време на провеждане на изследването. Изследователи са работили върху много алгоритми за корекция на движението, но до сега няма надежден метод за неговото разрешаване, забелязва се като размиване по ръбовете в сегментите на трансмитираното изображение.

За да се избегне неправилното тълкуване, е важно да бъде изяснен и да се познава поне произходът на тези артефакти.

В бъдеще се очаква, че рентгеновите прегледи могат да бъдат до голяма степен изместени от останалите по-сложни методи за извличане на изображения - ултразвукови, CT и MRI. Тяхното развитие е свързано с усъвършенстването на плоските детекторни панели с голям зрителен ъгъл и бързо отчитане. Очаква се да станат достъпни и за изграждане на 3D-изображения. Ще продължи подобряването на детекторите в посока намаляване дозите на радиация и получаване на изображения с голям контраст. Въвеждането на повече цвят в този метод също изчаква своето развитие.

Стандартните рентгенови образи и CT осигуряват анатомични изображения на основни органи, а SPECT, PET и fMRI [12] се използват за проследяване на функцията на органа. Ясно е, че би било изключително полезно, ако анатомичният и функционалният образ се обединят. Мултимодалните изображения се явяват като основна посока към подобряване на медицинската образна диагностика, т.е. комбинирането на анатомична томография (CT, MRI, ултразвук, дифузна оптична томография) и функционална томография (SPECT, PET). Към този момент стандартизираните МИ могат да бъдат напълно интегрирани в болничните

информационни системи. По отношение на лекарската намеса по време на предварителните настройки и режими за работа се търси решение за нейното намаляване в процеса на формиране на изображението и засилване на влиянието върху работния процес на дообработка на Отдели за образна диагностика.

6. Анализ на характера на грешката при различните модалности в получените образи

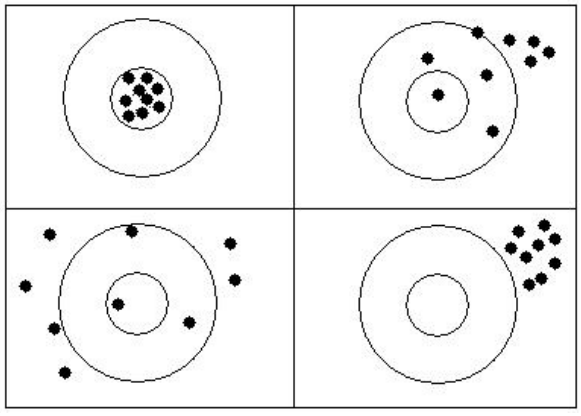
Анализът на грешките е проблем на всички научни изследвания и клинични услуги. Изображенията като данни включват известна неопределеност и се изисква техният статистически анализ. Изборът на технически фактори (кVp, mA, време в радиография) повлиява на появата на статистически колебания като *визуален шум и случайни колебания в стойностите на пикселите* в дигиталните системи [5,11]. Тези образи зависят от избора на размера на пиксела, степените на сивата скала и времето за получаване на образа, т.е. съществува непрекъсната промяна при задаването на тези и други променливи. Незадоволителна е „сигурността" на качествено получаване на образ на всички етапи в процеса на формирането му при ядрените и ехографските данни. За да се разчете даден образ, разчитащият специалист освен познания за методите на получаване трябва да е запознат и с вероятността на грешката и факторите, които я формират. Класифицират се две категории грешки: *определими и неопределими*.

1. Определимите грешки се получават в резултат на неправилно функциониране на апаратурата и непълна корекция на външни влияния. Това може да се намали при чрез допълнителни настройки за параметри, преценка на времето за експониране.

2. Неопределими грешки са грешки, които не могат да бъдат намалени чрез премахване или коригиране на външни фактори. Предизвикани са от първоначални неточности във величините, които се измерват, както и в процеса на получаване и визуализацията им.

Ако те са известни, може да се направи избор между различни теоретични вероятностни разпределения, чиито свойства са представени чрез математическа анализ.

За да се увеличи точността на резултатите, влиянието на двете категории грешки трябва да се намали. На фиг. 5. са посочени три понятия, които илюстрират графично надеждността при изпълнение на няколко комбинации при сериите от измервания [5].



Фиг. 5. а) точност, непредубеденост, прецизност. б) неточност, предубеденост, липса на прецизност. в) неточност, непредубеденост, липса на прецизност. г) неточност, предубеденост, прецизност.

7. Високо-специализирани иновативни техники

През последните години се предоставят технологични решения, които са по-удобни, безопасни и приемливи за пациентите. Те поставят сериозни предизвикателства пред разработчиците на самото устройство и разчитането на образа. Една от тях е ***безжично-***

пасивна капсулна ендоскопия [15, 16] и ангеография с наблюдение на целия път в реално време. Капсулата се движи пасивно през стомашно-чревния тракт с помощта на перисталтиката и предава снимки през целият път безжично. Резолюцията на изображенията е 256 x 256 dpi при кадър в секунда. Формулират се следните проблеми: 1. Капсулите на PillCam не могат да контролират своята позиция и посоката на движение. Този недостатък води до пропуски в изображенията. 2. Решението demosaicked-изображение е с ниска резолюция и важни региони могат да бъдат неволно пропуснати. В този смисъл се търсят решения, свързани с повишаване на резолюцията и едновременно с това, създаване на компресия, базирана на различната тежест на R-,G-,B-компонентите. 3. Времето за разчитане е повече от времето на заснемане. Проблемът е как да се извлекът само кадрите с данни, представляващи интерес. 4. Статистически са отчетени случаи на усложнения при преминаване на капсулата. За разрешаването на проблем с локализацията и, се използват CT или X-лъчи. Описаното технологични решение е един пример за потвърждаване на факта, че всеки нов метод в обработката на МИ е съпътстван от множество генерирани нови, които предстоят да се решават.

8. Екстерналните медицински образи като източник на контекст.

Медицинската фотография (екстернално заснемане) [25] е специализирана област на фотографията, обезпечаваща с документацията клиничното състояние на пациента. Тя е инструмент на невро-офталмолози, дерматолози, хирурзи; използва се за медицински и хирургически процедури, медицински изделия, проби от аутопсия и за целите на преподаването [2, 18, 23]. При проучване на Асоциацията по биокомуникации [26] на медицински фотографи, се установява, че около 90% са заети в университетски болници и изследователски центрове, а останалите - на свободна практика, работещи в клиники по дерматология, офталмология и пластична хирургия. За съжаление в България няма такава практика, както и не са обособени Отдели за образна диагностика.

Специализираните фотографски техники използват фотомикрография и ултравиолетова флуоресценция. Обекти на заснемане са: типовете кожа, лезии и морфологични аномалии на черепа, клепачи, кюнктива, част от торса, главата и ръцете (документиране на неврологични аномалии). Използват се конвенционални камери и биомикроскопски камери. Счита се, че при екстерналното статично заснемане, единствената променлива е развитието на проследения обект - рани, лезии, петна във времето. Всички останали условия трябва да останат същите - позициониране, осветление, режим, увеличение, перспектива, контраст [17]. Въпреки това, тези принципи на условна стандартизация следва да се прилагат за всеки набор от две или повече снимки, направени по различно време. Практически е изключително трудно да се стандартизират абсолютно тези променливи.

Смисълът на този вид проследяване можем да формулираме в две посоки:

1. Фиксиране на обекта чрез неговата локализация и характеристиките му както и неговото класифициране;

2. Проследяване на развитието на обекта чрез сравняване на статичните обекти в изображението.

Дискутират се в специализирани издания [24] въпросите за минимални, но достатъчни параметри за получаване на образ за целите на дерматологичната фотография. Някои автори [19] посочват, че в практиката на клиничната дерматология по отношение на резолюцията отдавна е достигнат необходимият минимум. Други автори [20] считат, че резолюция от 768x512 ppi (24 битов цвят) е достатъчна. Miot и неговите съавтори, предполагат, че резолюция на камерата от 1.3 Mp (1280x960 ppi) е подходяща за дерматологично проследяване [21]. Отчита се доколко по-високата резолюция би имала смисъл, дали да не бъде намален броят на пикселите и до каква степен да се дообработи образа, за да се

извлекат правилните граници, цвят и друга информация за него. Друг проблем е липсата на неуеднаквен стандарт RAW (8, 10, 12 бита на пиксел), предоставящ цялата информация, необходима за създаване на изображение. При него не се губи информация за заснетия обект, файлът е 2-6 пъти по-голям от JPEG файл, а последният се получава в резултат от стандартизирана компресия. Различни производители на техника използват свои стандарти: *.crw (Canon), *.ptx (Pentax) and *.nef (Nikon). В резултат на това конвертиране се получават снимки с различна степен на компресия и независимо от високата резолюция, се получава копие с много артефакти, загуба на фини детайли, лоши преходи в цветови порядък и светлосенките, което се визуализира като загубата на информация в различна степен.

Техниката, която отговаря напълно на гореизброените изисквания, са камери тип DSLR – digital single lens reflex. Използвайки дигиталните огледално-рефлексните камери, се прецизират следните *екстернални изисквания:*

1. Улавя се повече светлина, редуцират се част от шумовете, възможно е качествено изображение при лошо осветяване.
2. Наличието на ярка околна светлина при заснемане на обекта се компенсира.
3. Използване на RAW формат за изходни файлове, като следва допълнително конвертиране до JPEG с различна степен на компресия без да се губи информация, представляваща интерес.
4. DSLR използват универсални и специализирани обективи, светкавици и аксесоари към тях, филтри за разлика от компактните дигитални камери.

Налага се изводът, че екстерналното заснемане има своето място в медицинската диагностика, въпреки неглижирането му на фона на развитието на високо-специализираните иновативни техники. Въпреки безусловната разлика на добиването на образ чрез сложни методи, във втората част на алгоритъма – постобработката могат да се използват абсолютно идентични сегментационни методи, след които може да се пристъпи към сравняване и класифициране. Предимството на този метод е възможността за почти пълнен контрол над метаданните, генерирани от хардуера на полученото МИ.

9. Изводи

От направеното проучване могат да се формулират следните изводи, свързани с обработката на МИ:

1. Всяка от модалностите предоставя разултат, възможен за нейния обсег на генериране на данни.
2. Всяка от модалностите генерира няколко вида шум, които могат да попречат в една или друга степен за получаване на качествен образ.
3. МИ, независимо от начина на получаване, могат да бъдат подложени на постобработка за доизясняването на детайлите в образа. Трудно може да се прецени „степента на изясняване”, вземайки предвид наличието на артефактите и спецификите на човешката визуална система.
4. При извеждане на образа на монитор скалата от 4096 степени на сивото може да се визуализира само от специализирани монитори. Човешкото зрение също не би могло да разграничи тези яркости. Остава възможността системата за генериране на образ да работи с добре балансирани входни данни, за да разпознае и приложи очакваното от нея визуално сегментиране, изразено с използване на различен от сивия цвят, ако има такава опция в нейната модалност.

5. Иновационните технологични решения разрешават не до край определен проблем и поставят нови, най-често свързани с данни, от които не могат да се отделят тези, които ни интересуват.

6. Сложността на апаратурата и спецификата на човешката визуална система все повече убеждават анализаторите, че експертите-лекари не могат да се справят в очакваната степен с правилно разчитане на статичните образи и е необходимо въвеждане в практиката на болничните медицински заведения, на Отдели за образна диагностика.

7. Екстерналното медицинско заснемане е неглижирано. Липсват спезиализирани устройства за добиване на статични МИ за специфични кожни изменения. Специалистите-лекари дори с налични изображения нямат инструментариум, с който да сравнят настъпилите промени във времето. Разработките на тази тема са доста ограничени на брой [22]. Качеството им е спорно, тъй като са продукт на споделени впечатления или на фотографи, без да са разработили изисквания за целия обхват на условия за правилни входни данни или на лекари, използващи конвенционални дигитални апарати.

8. Липсват разработени методи с описание и стандартизиране на заснемане на отделните кожни изменеия по типове, тяхното класифициране и прилагане на постобработка при следене на изменеията, които настъпват, а именно:

- Фиксиране на обекта чрез неговата локализация и харатреристиките му – топография на обекта, линейни очертания и ръбове, обем, площ и цвят, както и неговото класифициране.

- Проследяване на развитието на обекта чрез сравняване на статичните образи в изображението, регистриране на промяна по гореизброените критерии.

Последните две точки ще бъдат обект на нашето по-нататъшно изследване.

Литература

[1]. Душков, Б.А., Королев А.В., Смирнов Б.А., Энциклопедический словарь: Психология труда, управления, инженерная психология и эргономика, Издательство: Академический проект, Деловая книга. Серия: Gaudeamus, 2005

[2]. Тагаев, Н.Н., Судебная медицина, Харьков „Факт”, 2003

[3]. Digital Imaging and Communications in Medicine (DICOM), 2011. DICOM Standard

[4]. Suetens, P., Fundamentals of Medical Imaging, Cambridge University Press, Cambridge CB2 8RU, UK, 2009

[5]. Hendee, W. R., E.R. Ritenour, Medicle Imaging Physics, Wieley-Less, Canada 2002

[6]. Peli, E., Contrast in Complex Images, Journal of the Optical Society of America A7 (10): 2032–2040. doi:10.1364/JOSAA.7.002032, Oct. 1990

[7]. Horowitz, P., W. Hill, The Art of Electronics, 2nd edition. Cambridge (UK), Cambridge University Press, UK,1989

[8]. Karthikeyan, K., C. Chandrasekar, Speckle Noise Reduction of Medical Ultrasound Images using Bayesshrink Wavelet Threshold, International Journal of Computer Applications (0975 – 8887), Volume 22– No.9, May 2011

[9]. Hsieh, J., Computed Tomography: Principles, Design, Artifacts and Recent Advances, Cambridge: Cambridge University Press, SPIE Publications, SPIE Monograph, Volume PM114, 2003

[10]. Liney, G., MRI in Clinical Practice, Springer-Verlag London Limited, 2006

[11]. Bankman, I.N., Handbook of Medical Image Processing and Analysis, 2nd Edition, San Diego, CA, USA, 2008

[12]. Srinivasan, L., R. Dutta, S. Counsell, J. Allsop, J. Boardman, M. Rutherford, A. David, Edwards Quantification of Deep Gray Matter in Preterm Infants at Term-Equivalent Age Using Manual Volumetry of 3-Tesla Magnetic Resonance Imagesp Pediatrics Vol. 119 No. 4 April 1, 2007 pp. 759 -765
[13]. Nishimoto S., An Vu, T. Naselaris, Y. Benjamini, B. Yu, Jack L. Gallant, Reconstructing Visual Experiences from Brain Activity Evoked by Natural Movies, Current Biology 21, 1641–1646, October 11, 2011
[14]. Constable R.T., L.R. Ment, B.R. Vohr, S.R. Kesler, R. K., Prematurely Born Children Demonstrate White Matter Microstructura Differences at 12 Years of Age, Relative to Term Control Subjects: An Investigation of Group and Gender Effects, Pediatrics 2008;121;306, 2007-0414
[15]. Lin, M., Study on Low-Power Image Processing for Gastrointestinal Endoscopy, VLSI Design, January 20, 2012
[16]. Rammohan, A, Naidu R. M., Capsule endoscopy: New technology, old complication. J Surg Tech Case Report 2011
[17]. Nayler, J. R., Clinical Photography: A Guide for the Clinician, J Postgrad Med 2003
[18]. Peres, M., The Focal Encyclopedia of Photography, 2007
[19]. Williams, R., Medical Photography Study Guide, MTP Press Limited, 1984
[20]. Siegel, DM, Resolution in digital imaging: enough already, Semin Cutan Med Surg 2002;21:209-15
[21]. Bittorf, A., M. Fartasch, G. Schuler, TL, Diepgen. Resolution requirements for digital images in dermatology. J Am Acad Dermatol 1997;37:195-8.
[22]. Miot, H.A., M.P. Paixγo, F.M. Paschoal, Basics of digital photography in Dermatology, An Bras Dermatol, 2006;81:174-80.
[23]. Rennert, R., M. Golinko, D. Kaplan, A. Flattau, H. Brem, Standardization of Wound Photography Using the Wound Electronic Medical Record, J Advances in skin and wound care, Vol.20, N1, 2008
[24]. http://www.delante.ru/catalog, 08.2013
[25]. http://www.ijo.in/
[26]. http://www.opsweb.org/?page=Externaleye
[27]. http://www.bca.org/publications/bca_news.html
[28]. http://94.23.199.161/site/index.php/en/solutions-en/external-modules-xt/articles-20-en
[29]. http://medcrunch.ru/oblachnoe-hranenie-medicinskih-izobrazheniy/

За контакти:
ас. Гергана В. Маркова
катедра „Компютърни системи и технологии”
ВТУ „Св. св. Кирил и Методий”
E-mail: gerganavioletova@abv.bg

проф. д-р Маргарита К. Тодорова
катедра „Компютърни системи и технологии”
ВТУ „Св. св. Кирил и Методий”
E-mail: marga_get@abv.bg

доц. д-р Мирослав Н. Гълъбов
катедра „Компютърни системи и технологии”
ВТУ „Св. св. Кирил и Методий”
E-mail: lexcom@abv.bg

ERP COST AND RETURN ON INVESTMENT CONDITIONS OF SUCCESS OF AN INVESTMENT

David Jessula

Abstract: Many firms are now aware of ERP (Enterprise resource planning) advantages. That IT system can gather all the enterprise functions with a unique database. Cost and Return on Investment must be integrated in firms strategies. A clear appreciation of conditions of success is essential.
Keywords: ERP (Enterprise resource planning), CMR (customer relationship management), SMR (supplier relationship management), ROI (Return on investment)

1. Introduction

Enterprise resource planning (ERP) is an IT system gathering all software modules processes of a company with a unique common database.

They are now integrated in many firms: big, middle or small enterprises. Best known in the IT market are SAP, Oracle, Sage, Cegid …

The software modules are independent : purchasing, procurement, inventory management, accounting processes…They can integrate the whole internal functions of the firm in a same and unique tool and enlarge external functions such as customer relationship, supplier relationship management and also e-business processes.

Using a workflow motor, when a data is put in the system, all the modules sharing the same database enters the data with a defined programmation.

All ERP suppliers can submit many proposals. The ERP market is getting larger and companies must adapt their strategy as to define best choices of investments.

Two questions are generally studied: what is the cost of an ERP implementation ? and What is the Return on investment?

2. ERP cost

Two sort of costs must be analysed:
ERP software costs
Other costs such as licence, maintenance etc

a) ERP software costs

Four steps belong to a progressive implementation:
Set up : définition of goals, means, needs and production of a migration plan
Detailed specification :research of the optimum adaptation of modules to the customer processes du client and to the operating modes
Realisation: dévelopment of spécifices, interface and data recover
Exploitation: operationnal production and necessary adjustments setting up

Cost can be appreciated in men/days
i,e how much time, in a dayly mesure, external interventors are necessary
With an average price of :
1000 euros by day for a project director
800 euros by day for a functional consultant
600 euros by day for a technical consultant

Detailed as:

1) Set up

Orientation study
Organisation study
Migration plan
Software installation
Formation of internal teams

- Project and pilotage structure
- General Scheme of implementation
- Integration quality Plan

	Project director	Fonctionnal Consultant	Technical consultant	Total
Total days	19	15	9	43
cost (euros)	19000	12000	5400	36400

2) Detailed specification:

By sector, process
Parametring
Maquetting
Adequation
Setting up Plan
Parametring file

- Fonctionnal deviations
- Testing
- Planning and charges

	Project director	Fonctionnal Consultant	Technical consultant	Total
Total days	12	65	10	87
cost (euros)	12000	52000	6000	70000

3) Realisation

Fiabilisation and data recover
Development spécifics, interfaces
Organisation of formation
Technical setting up
Testing and homologation

	Project director	Fonctionnal Consultant	Technical consultant	Total
Total days	26	43	54	123
cost (euros)	26000	34400	32400	92800

4) Exploitation

Site pilote implementation
Running in
Development

	Project director	Fonctionnal Consultant	Technical consultant	Total
Total days	10	16	13	39
cost (euros)	10000	12800	7800	30600

With a TOTAL :

	Project director	Fonctionnal Consultant	Technical consultant	Total
Total days	67	139	86	292
cost (euros)	67000	111200	51600	229800

Total software cost euros 229.800 can be appreciated as an average cost depending of the firm dimension, sector, specifications…
Total days are about 9-10 months, generally minimum

b) Other costs

Other costs can be evaluated as to be added to Euros 229.800 software cost

Software	229800
Licence	30000
Formation	75000
Servers	10000
Maintenance	35000
Interfaces	10000
Total	389800 €

Other costs may represent nearly half of the total ERP cost
Total cost may be appreciated linked with the company sector or other specifications

Examples:
Miro: French paper distributor
Global budget of 150.000€ including software, services and developments

Christofle: french luxury distributor
SCM Planning
New software (Logility Planning Solutions, American Software): Inventory Planning, Distribution Resource Planning, Demand Planning)

Investment: 110000 euros

Zhendre-froid industriel:
scheduling: 24000 euros
commercial management: 34 000 euros
inventory management: 46000 euros

Medecins du monde:
ERP Qualiac: modules Finance, Purchases, Stocks, accountings, budget, Web purchases
Cost 400000€ on 4 years

3. Return on investment ROI

Ernst Young : majority of firms do not calculate ERP ROI
Causes:
Difficulty to quantify ROI
Not compulsory
Charge or investment?

Quantitative criterias (costs reduction, profitability, financial results …) and qualitative ones (less time loosed, customer satisfaction, …) are generally integrated.

ROI of majority of firms:
Less than 2 years: 33%
From 2 to 3 years: 51%
From 4 to 5 ans:12%
More than 5 ans: 4%
84 % less than 3 years

Thomson
ROI positive 18 months after project initialization
ROI 80 millions euros trois years after

4. ERP Strategy

ERP Strategy has not always been a success
In a global strategy there is always an ERP strategy

SUCCESS AND FAILURES
Failures: AMR, Snap-on, Foxmeyer, Danone
Success: Avantis Pharma

AMR-Budget-Hilton Hotels: after 4 years and 125M$ spent on development, the project "Confirm" reservation system for hotel and rental car bookings crumbled in 1992

Snap-on (machinery, tooling): 50M$ lost sales, profit -22%: 3 years of design and implementation with the new order entry system Baan (1997)

FoxMeyer (pharmaceutical industry), has claimed that a bungled ERP installation from SAP et Accenture in 1996 helped drive it in bankrupcy. The case is still unresolved

Danone:
SAP end 2000 to get homogeneous processes in 120 countries and save 50M€ by year
Budget 150M€
3 trades: water, fresh milk products and biscuits: Evian,Volvic,LU (selling to Kraft Foods, buying Numico (nutrition))

WHAT ARE FAILURES CAUSES
Bad project planning and sheduling
Bad profitability analysis
General management insufficient support
Delays, budgets exceed
New technology inexperimented
Suppliers inable to respect their plannings
Human factors

SUCCESS
AVANTIS Pharma (Frankfort)
Développement of new supply chain concepts and implement of an Advanced Planning System (SAP APO)
Production planning reduction: from 4 months to 3 weeks
Inventory reduction :100M$
Cycle time reduction
Better customer service
Transport costs: - 10 %

What are Success conditions
A clear strategy
Optimize custumer service
Enforce cooperation partnerships
Global supply chain management
Security
New IT tools
Empowered employees
Benchmarking

An ERP implementation is always an investment. Many firms have investments plans largely more important that less than 0,5 million euros for an ERP. ERP strategy must be appreciated on ROI specially when quick success is obtained. Success conditions are above all qualitative.

Literature

[1]. BOWERSOX & CLOSS, Logistical Management: The integrated Supply Chain Process, Mac Graw-Hill ISBN
[2]. SHAPIRO R, HESKETT J.J, Logistics Strategy: cases and concepts, West Publishing
[3]. Philippe-Pierre DORNIER, Michel FENDER : La logistique globale, Editions d'Organisation
[4]. G.Baglin, O.Bruel,… Management industriel et logistique, Economica

For contacts:
David A. Jessula
E-mail: davidjessula@voila.fr

ИЗМЕРВАНЕ НА КАЧЕСТВЕНИТЕ ХАРАКТЕРИСТИКИ НА СОФУТЕРА

Александър Д. Димов

Резюме: Качеството на софтуерните системи придобива все по-голямо значение както за софтуерната индустрия, така и за изследователите в областта на софтуерното инженерство. Тъй като качеството е субективно понятие, се налага качествените характеристики да се формализират и да се дефинират подходящи метрики за тях. В тази връзка в статията се прави обзор на текущото състояние на метриките представителна извадка от известните качествени характеристики на софтуер.
Ключови думи: Качествени характеристики на софтуера, Софтуерни метрики.

Measurement of software quality characteristics

Aleksandar D. Dimov

Abstract: Quality of software systems receives more attention from both practicioners and academy in the area of software engineering. However quality is an abstract notion and have different perspective, depending on the users. In this respect it is needed to have formal measures for software quality and this paper makes a review of the state of the art in the area.
Keywords: Software quality characteristics, Software metrics.

1. Introduction

Software systems are getting an increasing impact in all areas of human life. This fact leads to a natural increase in the number of users and developers, which in turn raises the level of complexity in software systems. In order to manage the level of complexity in software systems, both researchers and practicioners in the area of software engineering should focus on software architecture. It is defined as [1] “the structure or structures of the system, which comprise software components, the externally visible properties of those components and the relationships among them”. It has also been noted that when designing the architecture of a software system, one should focus on the so-called quality characteristics toward such systems. Quality characteristics may have different definitions depending on the domain. For example, in terms of Service Oriented Architecture, they are refered as Quality of Service, which covers a wide range of techniques that match the needs of service requestors with those of the service provider's based on the network resources available [2]. In other domains quality requirements are called non-functional requirements and should be distinguished from functional requirements. The latter define what the system should do and the fomer put some additional conditions (in form of constraints or specifications) on how the system should perform or deliver its functionality. Throughout this paper we look at software system quality characteristics at a more abstract level and consider the related meanings as synonyms. Therefore we will refer to them just as system qualities.

Traditionally, quality of software is given a primary importance with respect to such application domains like embedded systems, safety critical sytems, real time systems and so on. However, in recent times, terms as *testability, maintainability* and others that look at the system, not during its execution, but during design time, gain bigger attention. Another interesting characteristic is system *usability* or also *learnability*, which look at the system from users point of view.

Although important software quality is still a general and quite abstract notion, which may have different meanings. One may ask the question “What does it mean that the system is usable?”.

For example, console interface may be regarded as more usable than graphical interface by advanced users, while beginner or intermediate users would prefer GUI. Another example is maintainability, which may be regarded as part of complexity, but may also have its own meaning, as we will see in next sections.

Currently there doesn't exist a universally accepted measurement framework of software quality, although there exist a lot of examples for software metrics in general [7]. A common approach for formalization of software quality is needed, which will establish unified measures for different quality characterstics of software systems. In such terms the goal of this paper is to study the current state of the art in measurement of software quality. This way we make the first step in unification of software quality characteristics.

2. Quality characteristics of software systems

Quality scenarios may be used to put in order different viewpoints in system quality. As defined in [1], a quality attribute scenario is a quality-attribute-specific requirement. It consists of six parts, as follows:

(1) Source of stimulus – this represents an entity (a human, a computer system, or any other actuator) that generated a stimulus, which make the software system to change its state.
(2) Stimulus – this represents a condition that needs to be addressed when it is detected by the system.
(3) Environment – the environment represents the combination of system state and the state and the condition of other actors which interact with it. For example, the system may be in an overload condition or may be running when a stimulus occurs, or some other condition may be true.
(4) Artifact – this may represent the whole system or some pieces of it that are stimulated.
(5) Response - this is the activity undertaken after the arrival of the stimulus, or it may represent a specific change in the system state.
(6) Response measure – this is the metric of the system quality. In other words, when the response occurs, it should be measurable in some fashion so that the requirement can be tested.

In order to be able to reason about quality, a formal definition of measures is needed, in the last part of quality scenario definition, given above.

Further in this section we present the current classification of quality characteristic of software systems and show their known measures. The list could be never complete and in this paper we focus on the following characteristics: testability, modifyability, maintainability, complexity, security, scalability, dependability.

Testability

Testability looks from the viewpoint of the efforts needed to perform system testing. Measures of testability may be used by different software development stakeholders, like project managers, testers and software developers. Popular metrics for testability are usually based on object-oriented characteristics of the system and include [3]:

- Depth of Inheritance Tree (DIT) – this is the size of the longest path of the class inheritance tree [4];
- Number of children – this is the number of classes that inherit from a given class;
- Number of methods – this is the number of the methods that a given class provides;

- Number of fields – this is the number of fields in the class.[1]

The common drawback of all these metrics is their orientation towards a single class and this way they are difficult to be generalized into a common metric for testability of a complex system as a whole. There may also exsist difficulties if the system is not designed following the object-oriented principles, for example in the case of legacy systems.s

Modifyability and maintainability

Modifyability and maintainability are similar quality characteristics and measure the level of ability of the system to change, due to corresponding changes in the user or environment requirements. Some options to measure modifyability and maintainability suggested in [5], include different measures for software complexity. For example the McCabe merics [8] for cyclomatic complexity or Halstead [9] complexity metrics may be used. The consideration is that if the complexity of the system is, higher, then it is more difficult to understand and hence – harder to modify and maintain. Other measures that may be used are simpler complexity metrics like lines of code, number of methods and again, DIT.

Although helpful complexity metrics have some disadvantages when used to measure modifyability and maintainability. For example, if the system is well documented, the maintainability may remain high, even for bigger code complexity values.

Dependability

According to Avizenis [10], dependability is defined as capability of a software system to deliver a service that can justifiably be trusted. Main threats for dependability are software errors which can be activated and finally cause a failure of the system. It is believed that a failure represents unwanted system behaviour when it no longer delivers a service as per the specification. Here we put special attention on dependability as it is characterized by a number of attributes, which are often recognized as separate quality characteristics. These attributes are as follows:

- Reliability – represents the belief that the system will be failure free for a given period of time;
- Availability – represents the degree of readiness of the system to deliver correct service;
- Confidentiality – means absence of unauthorized disclosure of information within the software system;
- Safety – means absence of catastrophic consequences on the user(s) and the environment even in case of failure;
- Integrity – represents absence of improper system state alterations;
- Maintainability – represents ability of the system to be repaired and updated.

Software reliability has a long history of research [11] and is usually measured as a probability of failure free software operation for a specified period of time in a specified environment. Althoug well defined and widely accepted, there exist the problem, with lack of models that makes possible correct evaluation of reliability. There exist two big groups of models for evaluation of reliability of software – black box and white box models. The first group of methods treats the software as a monolithic whole, i.e. as a black-box, and obtains a value for the reliability via statistical methods. The most common source of input data for such statistical methods is software testing. A common drawback of all black box reliability models is that they make specific assumptions that may lead to bias in the reliability estimate [12]. The group of white

[1] There also exist a number of additional metrics, described in [3], all of which are based on object oriented characteristic

box models take information about the internal structure of the software, in most cases based on the architecture [13, 14]. A drawback of white-box models is that they do not define how to obtain quantitative values in step (3). One possibility is to use the black-box models.

Formal definition of availability is more straightforward than reliability [1]. It may be defined as:

$$\alpha = \frac{\Delta t_f}{\Delta t_f + \Delta t_c}, \tag{1}$$

where α is software availability, Δt_f is the mean time between failures and Δt_c is the mean time to repair the system from the failure. However, there does exist different understanding of what availability is. For example in [15], two similar notions of availability and accessibility are defined.

The situation with other properties of dependability (except maintainability, which has been discussed above) is similar – there does not exist widely accepted formal definitions and models for safety, integrity, confidentiality, although some work exist in formalization of safety and security, as the latter is considered as a general notion which combines integrity, confidentiality and availability.

Safety

As stated above, safety – means absence of catastrophic consequences on the user(s) and the environment even in case of failure. A notable approach to measure safety is presented in [16], however it aims at evaluation not of the software system safety itself, but of the maturity in development process used to build a safe system. For instance, the approach proposes the following safety metrics:

- Percentage of software safety requirements, defined as the number of software safety requirements, divided by the number of software requirements.
- Percentage of software hazards, defined as the number of software safety hazards divided by the number of System Safety hazards.

The lack of metrics, aimed at evaluating safety of software systems and their constituent components, means that more work is needed in definition of safety measures and models for their formal evaluation

Security

Approaches for measurement of security use additional metrics. For example, the authors of [17] use several industry standards for definition and calculation of software security. A widely known list of vulnerabilities and their measures, together with scoring of vulnerabilities are used, both represented by industry standards. The vulnerability measures are:

- The distance between a possible attacker and the system;
- The complexity of the attack required to exploit the vulnerability once an attacker has gained access to the target system;
- The number of times an attacker must authenticate to a target in order to exploit vulnerability;
- Different impacts of a successfully exploited vulnerability.

The security of the software is represented by the following equation:

$$Sec = \sum_n (P_n \times W_n), \tag{2}$$

where W_i is the weight of the i^{-th} software weaknes, which is defined by the vulnerabilities that activate it and P_i represents the risk of the corresponding weakness.

3. Conclusions

The paper makes a brief overview of some of the currently known measures for different software quality characteristics. The presented here quality characteristics may not be complete but claims to be representative, as it shows some of the most widely used ones. Table 1 summarizes the facts, listed in section 2 about software quality measurement.

Table 1: Comparison of software quality characteristic measures

Quality characteristic	Measure	Formal models for evaluation
Testability	Object oriented metrics (OOM)	N/A
Modifyability and maintainability	OOM and complexity measures	Most of them are deprecated
Dependability	Percentage. Metrics are not available for all dependability properties	A lot of models mainly for reliability and availability. Howevere, most models have serious limitations for practical application.
Safety	Available only for development process, not for the software itself.	N/A
Security	There exist some metric	Yes

A deeper analysis of most measures shows that they are abstract and more work is needed in the area of software quality measurement. As seen in table 1 directions for future research include refinement of measures as well as work on elaboration of formal models for their calculation. Additionally for many characteristics like usability, adaptability and etc. there doesn't exist sound enough measures.

Acknowledgements

The research, presented in this paper was partially supported by the research fund of University of Sofia "St. Kl. Ohridski", under contract No. 107/19.04.2013.

References

[1]. Bass, L., P. Clemens and R. Kazman. Software Architecture in Practice, Addison Wesley, 2013.
[2]. Anbazhagan, M. and A. Nagarajan. Understanding quality of service for Web services. 2002. <http://www.ibm.com/developerworks/webservices/library/ws-quality/index.html>. Last visited, August 2013.
[3]. Bruntink, M., and A. Deursen. Predicting class testability using object-oriented metrics. Source Code Analysis and Manipulation, 2004. Fourth IEEE International Workshop on. IEEE, 2004.
[4]. Shatnawi, R. A quantitative investigation of the acceptable risk levels of object-oriented metrics in open-source systems. Software Engineering, IEEE Transactions on 36.2 (2010): 216-225.

[5]. Land, R. Measurements of software maintainability. Proceedings of ARTES Graduate Student Conference, ARTES. 2002.
[6]. Wang, J., et al. Security metrics for software systems. Proceedings of the 47th Annual Southeast Regional Conference. ACM, 2009.
[7]. Fenton N. Software Metrics — A Rigorous Approach. Chapman & Hall, London, 1991.
[8]. McCabe T. A complexity measure. IEEE Transactions on Software Engineering, SE-2,1976, pp. 308—320
[9]. Halstead M.H. Elements of Software Science. New York, Elsevier, 1977.
[10]. Avizienis, A., Laprie, J-C., Randell, B.: Basic concepts and Taxonomy of dependable and secure computing. IEEE Trans on Dependable and Secure computing. Vol. 1, No. 1.
[11]. Lyu, M. Handbook of software reliability engineering. Vol. 3. CA: IEEE computer society press, 1996.
[12]. Chandran, S. K., Dimov, A., & Punnekkat, S. (2010, June). Modeling uncertainties in the estimation of software reliability–a pragmatic approach. In Secure Software Integration and Reliability Improvement (SSIRI), 2010 Fourth International Conference on (pp. 227-236). IEEE.
[13]. Dimov, A. and Punnekkat, S. On the Estimation of Software Reliability of Component-Based Dependable Distributed Systems. Proc. of the 1st International Conference on the Quality of Software Architectures (QoSA 2005), LNCS 3712. Springer-Verlag, September 2005, Erfurt, Germany, pp. 171-187.
[14]. Goseva-Popstojanova and K. Trivedi. Architecture Based Approach to Reliability Assessment of Software Systems, Performance Evaluation, Vol.45/2-3, June 2001.
[15]. OASIS Web Services Quality Factors Version 1.0, Committee specification 01. available at <http://docs.oasis-open.org/wsqm/WS-Quality-Factors/v1.0/cs01/WS-Quality-Factors-v1.0-cs01.pdf>. Last visited – August 2013.
[16]. Kumar, S., P. Ramaiah, and V. Khanaa. Identification of Software Safety Metrics for Building Safer Software based Critical Computing Systems. International Journal of Computer Science and Application (2010).
[17]. Wang, J. A., Wang, H., Guo, M., & Xia, M. (2009, March). Security metrics for software systems. In Proceedings of the 47th Annual Southeast Regional Conference (p. 47). ACM.

За контакти:
доц. д-р Александър Димов Димов
катедра „Софтуерни технологии”
Софийски университет „Св. Климент Охридски”
E-mail: aldi@uni-sofia.bg

СРАВНИТЕЛЕН АНАЛИЗ НА МЕТОДИТЕ ПРИ РАЗПОЗНАВАНЕ НА ЛИЦА

Петя П. Петрова

Резюме: Методите за разпознаване на лица са различни по идеята си, но взаимосвързани в йерархична структура. Всеки следващ променя бързодействието и качеството в разпознаване в положителна посока. В статията е направен анализ на съвременните методи за разпознаване на лица в контекста на техните предимства и недостатъци.
Ключови думи: разпознаване, методи в разпознаването, точност на разпознаване.

Comparative analysis of methods for face recognition

Petya P. Petrova

Abstract: There are different methods of face recognition. Although, they are different in concept, they are interrelated in structure with changes that boosts the quality of recognition. This should be connected to the last sentence. This article analyses the current methods of face recognition with their advantages and disadvantages.
Keywords: Recognition, methods of detecting, recognition accuracy

1. Увод

При наличие на различни алгоритми и методи за разпознаване на изображения, типичният метод за разпознаване на лица се състои от три компонента. Преобразуване на изходното изображение към първоначалния облик (преработка или математически преобразувания, например изчисление на главният компонент). Разделяне на ключови характеристики - вземат се първите n главни компоненти, относно дискретното косинусово преобразувание. Третият е механизъм на класификация или моделиране - клъстерен модел, метрика, невронни мрежи и т.н. Изграждане на методи за разпознаване се базира на априорна информация за предметната област - характеристики на човешкото лице в частност и регулиране на експерименталните данни, появяващи се в хода на разработка на методите.

В [1] сред различните подходи за разпознаване на образи се формулира статистическият подход. Скоро невронните мрежови техники и методи, внесени от статистическата теория за обучение са заслужили внимание. Дизайнът на системата за разпознаване изисква задълбочено внимание върху въпроси като: модел класове със следене на средата, модел на представяне, извличане на отделна част и избор, клъстеризиран анализ, класификатор за проектиране. Глобалният проблем в разпознаването на сложни модели с произволна ориентация, място и мащаб, остава неразрешен.

2. Принципен компонентен анализ (Principal component analysis - PCA)

Предимства на PCA

PCA е специфичен случай на факторен анализ, полезен в анализа на дълги серии във времето. Служи за различни допускания относно основната структура и определя собствени вектори за различни малки матрици. По-добър е от по-ранните двумерни използвани статистически техники. Изследват се вътрешните взаимодействия между набор от променливи, причинени от общи тенденции. PCA е начин за идентифициране на модели в

данните, а също така и за изразяване на данните по такъв начин, че да подчертаят техните прилики и разлики.

PCA се ползва за компресиране на данни чрез намаляване броя на размерите, без съществена загуба на информация. Основните компоненти са такива, че всеки следващ обяснява един максимум от останалите вариации. Първият компонент е такъв, че да обясни максималният процент на общото разсейване, а вторият компонент да обясни повечето от останалите вариации. PCA е напълно непараметричен (plug-and-play): всеки набор от данни може да бъде включен към отговора, неизискващ никакви параметри за коригиране и без оглед на това как данните се записват. Може да се счита за положителна функция.

Ограничения на PCA

PCA е нестатистически метод и няма вероятностно разпределение. PCA представя данните в опростена и редуцирана форма. Новите променливи са линейни комбинации на оригиналните променливи. В допълнение, за да работи PCA

$$z_I = \frac{x_i - \mu_x}{\sigma_x} \tag{1}$$

точно, се използват стандартизирани данни. Като средната стойност да е нула и обективната оценка на вариацията да е единица:

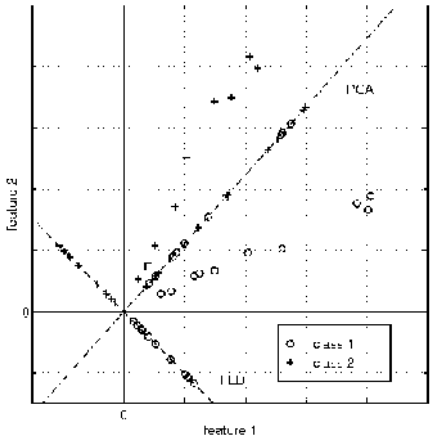
$$z_i = i^{th} \tag{2}$$

където $z_i = i^{th}$ е стандартизирана променлива.

Стандартизиране - резултатите на PCA са относно нестандартизирани променливи. Поради това анализирането на PCA и приложението му е по-трудно. PCA се свързва с освобождаване от взаимосвързаност и взаимозависимост на променливи. Методът практически е неприложим при анализ с различни условия на осветеност. Първите основни компоненти отразяват промените в осветлението. Развиват се изображения, сходни с нивото на светлината.

3. Линеен дискриминантен анализ на Fisher (Fisher's Linear Discriminant-FLD)

Проекционните пространства на изображенията в LDA (линеен дискриминантен анализ на Fisher [2, 3], LDA) пространствената функция минимизира вътрекласовото, както и максимизира междукласовото разстояние на пространствените признаци. Примерът проектира в пространството характеристики за два класа чрез PCA и FLD - фиг. 1. Слабост е, че PCA е агностичен към източника на данни. Подходът използва идеализирани условия - глобални настройки на светлината, неутрално изражение на лицето, липса на смущения като очила и бради (невъзможно да се постигне чрез предварителна обработка). Проекциите на фиг. 1 са осъществени чрез PCA и FLD.



Фиг. 1. Проекции в пространството за два класа

4. Разпознаване с използване на класове специфични линейни проекции

В [2] тренировъчната серия съдържа лица с няколко условни условия на осветеност. Чрез линейна комбинация може да се получат други условия на осветеност. Забелязана е висока точност на откриване (около 96%) за широка гама от условията на осветление, различни изражения на лицето и наличието или отсъствието на очила. Както и ниска разпознаваемост на метода на собствени лица в аналогични условия. Много въпроси са загадка. Методът се прилага в голяма база данни. Ако не се променя ъгълът, експериментите се провеждат с промяна на осветлението, без да се променят други фактори. Дали методът ще бъде ефикасен при съчетание е неизвестно. Необходима е предварителна качествена обработка. След това изображенията се превеждат към стандартни условия. Методът се базира на линейна разделимост на класовете в пространството на изображението.

5. Автоматична интерпретация и кодиране на лицеви изображения с използване на гъвкави модели (Automatic Interpretation and Coding of Face Images Using Flexible Models)

Лицевите изображения в [4] са трудни за тълкуване и силно променливи. Източници на променливост - външен вид, 3D поза, изражение на лицето, както и осветление.

Фиг. 2. Основни видове вариации на сиво

Проблем е първият режим на вариации, 80% от вариациите на сивото. Другите възможни вариации са блокирани. Извършва се на основата на контурите на лицата. Извличат се чрез линии на главата, ушите, устните, носа, веждите и очите. Контурите са ключови позиции, между които позицията на точките, които принадлежат на контура, са изчислени чрез интерполация. Ключовите точки са поставени ръчно в набор от тренировъчвъчни изображения. Извличане на информацията за интензивността на пиксела е лежаща на линията, перпендикулярна на контура за всяка точка от контура. При търсенето на контурите на ново лице, симулацията с целева функция е от две съставляващи. Първата се увеличава с интензивността на пикселите, извлечени от перпендикулярния контур на линиите. Втората е съвпадение на контура с формата на контура на примера. Екстрахира се контур на чертите на лицето. Контурният модел използва полутонов модел. Повишава точността при разпознаване. Предизвикателство е правилното разпределение на тези контури. Използването на този метод в разпознаването е недостатъчен.

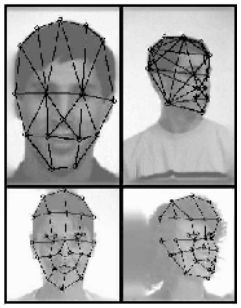
6. Сравнение от еластична графика

В статии [5], [8] човек е представен като графика, чиито върхове са разположени в ключови точки на лицето, като контурите на главата, устните, носовете, и техните крайни точки. За всяка повърхност е маркирано разстоянието между върховете. Във всяка точка се изчисляват коефициентите на функциите на Габор за петте различни честоти и осемте

различни ориентации. Набор от такива коефициенти $\{J_i\}$ се нарича (jet). Те характеризират местните части на изображението и се използват за две основни задачи. За намиране точки, съответстващи на предварително определена зона, на две различни изображения. За сравнение на две съответстващи зони от различни изображения. Всеки коефициент

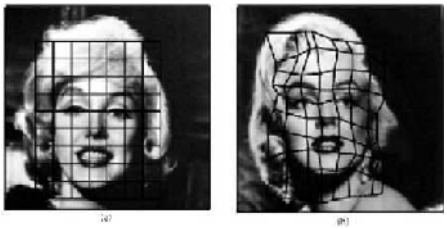
$$J_i = a_j \exp(i\varphi_j) \tag{3}$$

за точките от една област на различни изображения се харектеризира с аплитуда a_j, която бавно се променя с изменението на точките и фазата φ_j, която се върти със скорост, пропорционална на честотата на вълновия вектор от основния уейвлет. За получаване на бързо и надеждно сближаване при търсене с помощта на прости модели като например дифузия. По-съвършените функции на сходство включват информация за фазата.



Фиг. 3. Еластична графика, обхващаща портретното изображение

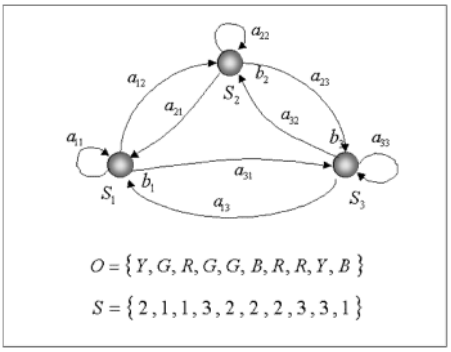
Подобрение на метода е прилагането на линейни преобразовния на jet за компенсиране промяната на ъгъла [6]. В по-ранна версия не се използват първоначалните определени ключови точки и структурни графики [7], [8]. Някои от тях ползват множество jets (фиг. 4). В неизвестно изображение са намерени точки на съответствие. Намерени са точки, които се основават на изкривената мрежа. Измерва се някаква мярка за нейното нарушаване, която се отнася за измерване на най-много подобни образи.



Фиг. 4. Гъвкава решетка, насложена върху изображението и нейното изкривяване

7. Разпознаване чрез скрит модел на Марков (HMM) и псевдо двумерни скрити модели на Марков (P2DHM)

Методът [14] намалява изчислителната сложност на предишния HMM, като се запазва същата скорост на разпознаване. Методът HMM е мощен инструмент за моделиране на различни процеси и разпознаването на образи. Моделът позволява да се отчитат непосредствено пространствено-времевите характеристики на сигналите и затова се използват в последно време и в разпознаването на изображения (в частност лица).



Фиг. 5. Диаграма на HMM за поредица от O наблюдения и последователни членове S

Всеки модел на фигурата е набор от N-членове, на фиг. 5 горната част от графиката

$$S = \left(S_{1'}, S_2, S_N\right), \tag{4}$$

между които са възможни преходи (на фигурата дъги) - [13]. Във всеки момент системата е в много специфично състояние. Първите модели на Hidden Markov имат следния смисъл. Следващото състояние зависи само от текущото състояние. Преходът във всяко състояние се генерира от наблюдавания символ и съответства на физическия сигнал на изходната симулирана система. При реални процеси последователността от състоянията е скрита от наблюденията. Известна е изходната система, последователността от наблюдаваните символи,

$$O = O_1 O_2 ... O_T, \tag{5}$$

където всяко наблюдение O_t - символ на V и T - броят на наблюденията в последователността. Разпознаването се извършва чрез скрити модели на Hidden Markov.

Има няколко целеви въпроса:

1. При поредица от наблюдения $O = O_1 O_2 ... O_T$ и потребителски модел

$$\lambda = \left(A, B, \pi\right), \tag{6}$$

как се оценява вероятността

$$P\left(O / \lambda\right) \tag{7}$$

от образуването на този модел поредица от наблюдения? Нарича се разпознаване на образи.

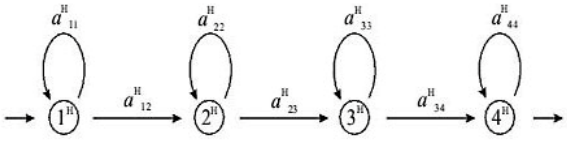
2. При съществуването на поредица от наблюдения $O = O_1 O_2 ... O_T$ и потребителски модел $\lambda = \left(A, B, \pi\right)$, как да се избере последователността от състоянията

$$Q = q_1 q_2 ... q_T \ , \tag{8}$$

за да са оптимални (необходими за последваща корекция на параметрите на модела)?

3. Какви образи коригират параметрите на модела λ, за максимизиране на $P\left(O / \lambda\right)$? Моделът да съответства повече на неговия клас, едно от изображенията, на които е дадена последователност от наблюдения.

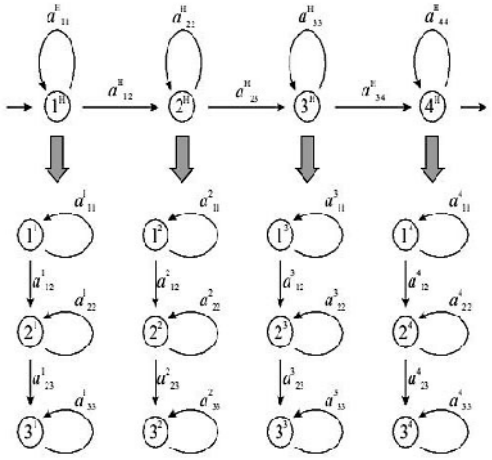
Първият проблем има точно аналитично решение - процедура на директния обратен преход. Останалите нямат точни аналитични решения. За втория проблем решение е алгоритъмът на Viterbi, а за третият - алгоритъм на Baum-Welch. Всяко състояние има само едно следващо. Възможно е плавно постъпване в същото състояние. Модели като този в [13] са с широко приложение в разпознаването на изображения.



Фиг. 6. Линеен скрит модел на Марков

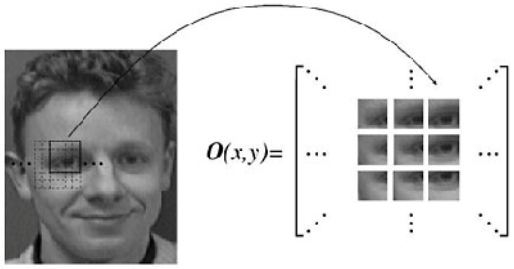
Двумерният модел на Марков може да симулира моделиране изкривяване на изображението.

Взаимните разположения на участъците не са отделени хоризонтално или вертикално, а са в двете посоки едновременно. Чрез P2D-HMM се намалява изчислителната сложност. Състои се от няколко линейни вертикални модели на вертикално ниво и линеен хоризонтален модел на горното ниво на входа, на който постъпват изходните модели на ниско ниво – фиг. 7. Всяко състояние на модела с най-високо ниво включва поредица от състояния на съответстващия модел на по-ниско ниво. Моделите от по-ниско ниво не са свързани по между си. Първоначално [11] най-високото ниво на моделите е вертикално. В [15] моделите от високо ниво са направени хоризонтални. Вертикалният модел на по-ниско ниво е съобразно с това-очите да могат да се разполагат на различни височини. При псевдодвумерния модел може да се разгледат локалните деформации и взаимното разположение на участъците в изображението. Отчита се характера на деформацията.



Фиг. 7. Псевдо-двумерен скрит модел на Марков

Подадените на входа HMM са квадратни участъци от изображения - фиг. 8. Частите се екстрахират с 75% припокриване един с друг, което дава най-добра точност при разпозпознаването. При HMM има възможност за сегментирано разпознаване на изображение. То е последоветлност от състояния, фиг. 8 [9].



Фиг. 8. Отстраняване на участъци от образец

В [9], като начална инициализация всеки модел използва всичките изображения от предварителната подготовка. В [10] са направени експерименти в разпознаването на компресирани изображения. Била е достигната 99,5% точност в същата база. Недостатък на HMM е, че не разполага с дискриминация. В [12], за да се определи дали дадено лице се съдържа в подготовката, е създаден алгоритъм на вероятностите. Първоначалната подредба се образува от тези модели. За непознато изображение моделите са подредени по вероятност на отговора на неизвестното изображение. Голяма стойност на отклонение от първоначалната подредба показва, че изображението принадлежи към класа на неизвестните лица.

8. Заключение

PCA е факторен анализ, който е полезен при дълги серии във времето. При автоматичната интерпретация и кодиране на лицеви изображения с използване на гъвкави модели недостатък е, че изображенията са деформирани до средна форма и сиво ниво. При сравнение от еластична графика човек е представен като графика, като се ползват контури. Гъвкава решетка, насложена върху изображението и нейното изкривяване, илюстрират по-ранна версия за разпознаване. HMM е инструмент за моделиране на различни процеси като разпознаване на образи. За намаляване на изчислителната сложност са приложените псевдо-двумерни модели на Марков, където се отчита същността на деформацията.

Методите за разпознаване се допълват. Всеки следващ дава нова насока и идея за предимство пред предходния и поне един проблем за решаване, предизвикателство за по-нататъшните експерименти. Всеки има и предимства и недостатъци. Предпоставка за по-добър е времето на появата им тъй като технологиите са властващата доминантна сила, която контролира тяхното развитие. Връх в разпознаваемостта като точна модификация за най-добър метод няма, понеже развитието на методите и алгоритмите за разпознаване продължава и до днес. Процентът в разпознаваемостта е важен, но не и определящ в сравнението на методите. Обект на изследване и интерес е набор лица в присъствие на смущения като промяна на коса, наличие на брада и очила. И към настоящият момент все още има много предизвикателства в тази насока.

Литература

[1]. Robert P.W. Duin and Jianchang Mao (Senior Member,IEEE)November 1,1999 . Statistical Pattern Recognition: A review Anil K.Jain(Fellow,IEEE)*

[2]. Belhumeur P. N., Hespanha J. P. and Kriegman D. J. Eigenfaces vs Fisherfaces: Recognition Using Class Specific Linear Projection. IEEE Transactions on Pattern Analysis and Machine Intelligence 1997, Vol. 19, pp. 711-720.

[3]. Hallinan P. L., Gordon G. G., Yuille A. L., Giblin P., Mumford D. Two- and Three-Dimensional Patterns of the Face. Natick:A. K. Peters Ltd. 1999. - 260 p.

[4]. Andreas Lanitis, Chris J. Taylor, and Timothy F. Cootes, IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 19, NO. 7, JULY 1997 743.Automatic Interpretation and Coding of FaceImages Using Flexible Models

[5]. Wiskott L., Fellous J.-M., Krueger N and Malsburg C Face Recognition by Elastic Bunch Graph Matching.IEEE Transactions on Pattern Analysis and Machine Intelligence 1997, Vol. 19, pp. 775-779

[6]. Krueger N.An Algorithm for the Learning of Weights in Discrimination Functions Using a Priori Constraints. IEEE Transactions on Pattern Analysis and Machine Intelligence 1997, Vol. 19, pp. 764-768.

[7]. Wurtz R. P.Object Recognition Robust Under Translations, Deformations, and Changes in Background. IEEE Transactions on Pattern Analysis and Machine Intelligence 1997, Vol. 19, pp. 769-775.

[8]. Grudin M. A., Lisboa P. J., Harvey D. M.Compact multi-level representation of human faces for identification. Image Processing and its Applications, pp. 111-115, 1997.

[9]. Eickeler S., Muller S., Rigoll G.High performance face recognition using Pseudo 2-D Hidden Markov Models // Gerhard-Mercator-University Duisburg, Germany, 1998. - 6 p.

[10]. Eickeler S., Muller S., Rigoll G.Recognition of JPEG Compressed Face Images Based on Statistical

[11]. Samaria F.Face Recognition Using Hidden Markov Models // PhD thesis, Engineering Department, Cambridge University, 1994.

[12]. Eickeler S., Jabs M., Rigoll G.Comparison of Confidence Measures for Face Recognition // Gerhard-Mercator-University Duisburg, Germany, 2000. - 6 p.
[13]. Rabiner L. R.A tutorial on Hidden Markov Models and selected applications in speech recognition.Proceedings of the IEEE, 1989. - Vol. 77(2). - P. 257-285.
[14]. Ara V.Nefian and Monson H.Hayes III .Center for Signal and Image Processing, School of Electrical and computer Engineering Georgia Institute of Tehnology , Atlanta,GA 30332, Hidden Markov for fase recognition
[15]. Gerhard-Mercator-University Duisburg, Germanyg Department of Computer Science
Faculty of Electrical Engineering. High performance fase recognition using pseudo 2-D Hidden

За контакти:
инж. Петя П. Петрова
Катедра КТТ
Технически университет - Варна
E-mail: p5ia@abv.bg

Bulgaria Communications Chapter

КОМПЮТЪРЕН МЕТОД ЗА БЕЗКОНТАКТНО ОПРЕДЕЛЯНЕ НА ПСИХОЛОГИЧЕН СТРЕС В ГОВОРА

Петър С. Апостолов

Резюме: В статията е разгледан проблемът за използване на компютърен метод за определяне на психологичен стрес в говорна информация. Направен е преглед на съществуващите анализатори на стреса, използвани в практиката на правораздавателните органи. Предложена е програма на Matlab за компютърна обработка на звукова информация. На основата на цифрово интегриране са предложени емпирични количествени критерии за оценка на стреса. В допълнение е приложен метод за анализ на звукови сигнали с трансформация на Хилберт-Хуанг. Показани са примери за анализ на информационни сигнали, направени успоредно с полиграфско изследване в Института по психология на МВР.
Ключови думи: полиграф, психологичен стрес, Хилберт-Хуанг трансформация

Computational methods fot uncontactless determination of psychological stress speech

Petar S. Apostolov

Abstract: In this paper the problem of using computer-based method for the determination of psychological stress in speech information is discussed. A review of existing stress analyzers used in the practice of law enforcement agencies is made. A Matlab program for computer processing of speech information is proposed. On the basis of the numerical integration empirical quantitative criteria for evaluating the stress are proposed. In addition the method is applicable for the analysis of sound signals by a Hilbert-Huang transformation. There are examples of the analysis of data signals carried along with polygraph examination at the Institute of Psychology of the Ministry of Interior.
Keywords: polygraph, psychological stress, Hilbert Huang transformations

1. Увод. Полиграфски метод за верификация на истината

Полиграфските изследвания установяват дали изследваното лице казва истината или лъже. В правоохранителнте и правораздавателните институции тази диференциация е от особено важно значение. Идеята за създаване на научен подход при полиграфските изследвания се появява през 20-те и 30–те години на XX век.

През 1948 г. Клив Бакстър създава и оглавява звеното на ЦРУ за разпити, което използва детекцията на лъжата, като основен метод на работа. Той е основател на подхода на цифрово оценяване или квантитативния подход в полиграфските изследвания. Посредством прикрепени датчици към тялото на изследваното лице се отчитат, а в последствие и оценяват настъпилите физиологичните промени в кожно-галваничното съпротивление, дихателните движения, в кръвното налягане и пулса, когато изследваното лице изрича лъжа на зададените въпроси от конкретния формат тест избран за съответния казус. Тези физиологични процеси не могат съзнателно да бъдат контролирани от изследваното лице, тъй като регулирането се осъществява от автономната нерва система.

Полиграфските изследвания имат висока валидност и надеждност. Във връзка с това твърдение, може да се посочи и обемно изследване на Н. Енсли от 1990 г., където се разглежда валидността на полиграфския тест при реални случаи. Енсли използва данни от проведени полиграфски изследвания в различни страни - САЩ, Канада, Япония, Израел и Полша. На базата на разгледани 2042 случая, за които има реални данни за потвърдена вина или невинност, валидността е 98% [1].

Близка до научната концепция на полиграфа е тази на описаното в настоящата статия технически средство, наречено звуков анализатор на стрес, с което се извършват подобни аналитични действия, но по безконтактен начин. В основата на действие на уреда е звуковият анализ на човешката реч. В литературата е прието, че когато изследваният обект говори неистина, има основание да се счита, че лицето е в психологичен стрес. Следователно отчитането на стреса в говора е един вид безконтактен „детектор на лъжата".

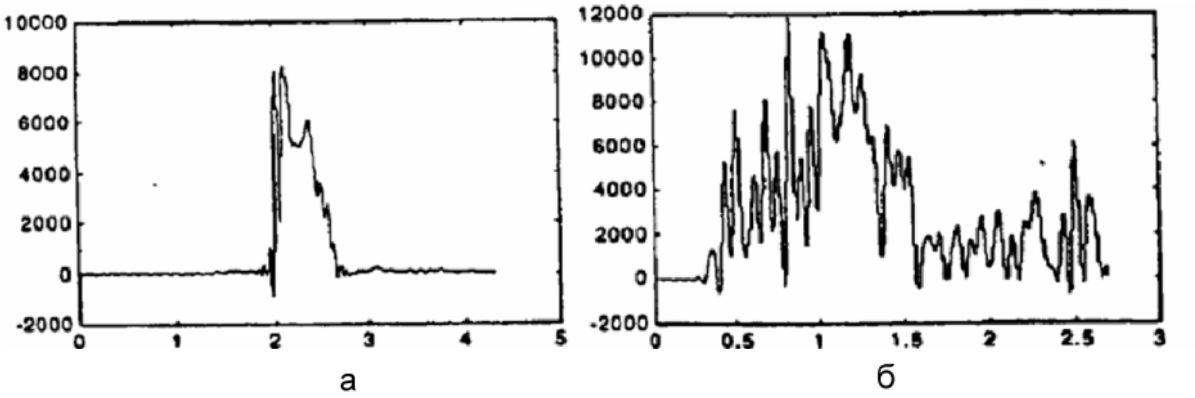
2. Теоретични основания. Преглед на съществуващите методи и технически средства

Успоредно с полиграфските изследвания, в много страни се работи за създаване на безконтактни прибори за установяване на достоверността при разследвания на престъпни деяния от правоохранителните органи. Една част от тях са свързани с изследвания на човешкия говор и по-точно на психологичния стрес в гласа.

Научният фундамент се базира на откриването на гласовия микро-тремор през 50-те годинина на 20 век. През 1971 г. Олоф Липолд, изследовател в Лондонския университет, публикува резултатите на изследването, което е започнал през 1957 г. в списание Scientific American. В статията на Липолд [2] се дефинира понятието „психологичен тремор". Авторът използва изследванията на явлението от Мартин Холидей и Джо Редфийм, проведени десетилетие преди това в Националната болница в Лондон. Те открили, че контракцията на волеви мускул е съпроводена с тремор под формата на осцилации. Според Липолд, всички мускули в тялото, включително гласовите струни, вибрират в диапазона от 8 до 12 Hz. Авторът счита, че треморът е функция на сигналите към и от двигателните неврони и че е аналогичен на самонастройваща се серво система със затворен цикъл. Мускулите се свиват и отпускат, като по този начин поддържат постоянно напрежение, подобно на мускулен тонус. В моменти на стрес тялото на индивида се подготвя за отбрана (битка) или бързо отстъпление (бягство), като увеличава готовността на мускулите за действие. Техните вибрации се увеличават от нормалния диапазон 8 до 9 Hz до стресиращия 11 до 12 Hz. Тези трептения се включват в човешката реч, като модулират говора. Тези модулации не могат да бъдат доловени от човешкото ухо, но съществуват електронни прибори, които ги установяват с висока точност.

Статията на О. Липолд третира медицински аспекти и никъде не става дума за използване на психологичния тремор за верификация на истината. Тези научни резултати карат много учени и инженери да конструират редица технически средства за създаване на звуков „детектор на лъжата". Изобретен е метод за детектиране на психологичния стрес с оценка на психологичните промени в човешкия глас, който е защитен с Патент - US Pat. № 3971034 на името на Бел. Гласовите модели на стрес първо са представени като метод за измерване на психологичния стрес, като се започне от «Устройство за оценка на психологичния стрес» (PSE – Psychological Stress Evaluator), проектирано от Форд, Бел и МакКистън през 1971 г. Уредът PSE записва в графичен вид диаграма върху лента от хартия, подобно на електрокардиограма. Устройството е магнетофон с променлива скорост на възпроизвеждане на звука. След запис, скоростта на лентата се забавя 8 пъти и сигналът се пропуска през филтри с различно затихване и се анализира. Всяка реакция в записания говор произвежда вертикални движения на писеца, които се анализират и определят субективно като стресови или не- стресови модели (шаблони). За критерии служат дължината и формата на шаблона. Общо казано, оценяваните гласови шаблони са във всякакви форми и размери. Те включват разлики за мъже и жени, регионални и международни лингвистични разлики и др. Независимо от това, моделите на човешки глас имат много общи характеристики и подобни отличителни черти, свързани със стреса, а именно, че тенденцията при стрес е да се формира шаблон с приблизително правоъгълна форма. В такъв случай се говори за т. нар.

тенденция «блокиране». На фиг. 1а е показан шаблон при произнасянето на думата “fix” при липса на психологичен стрес. Шаблонът има приблизително триъгълна форма.



Фиг. 1. Шаблони на думата “fix” при отсъствие (а) и наличие (б) на психологичен стрес

На фиг. 1б е показан същият шаблон при наличие на психологичен стрес. Формата наподобява правоъгълник.

В последните години са създадени редица компютъризиран средства. В тях се използват различни способи, заимствани от теорията на цифровата обработка на сигналите. Оценката на стреса се определя от нивото на мощността на сигналите в конкретната честотна лента. Специален интерес представлява създаденият в Националния институт по удостоверяване на истината NITV (National Institute for Truth Verification) компютъризиран анализатор на стрес CVSA (Computer Voice Stress Analyzer). Ръководител на проекта е Чарлз Хъмбъл, д-р. CVSA представлява преносим компютър с микрофон и вграден софтуер за визуална и количествена оценка на стреса. Методът е защитен с патент № US 2005/0131692 A1 под името *„Method for Quantifying Psychological Stress Levels Using Voice Pattern Samples”*[3]. На основата му е създаден FACT алгоритъм за оценка. Претенциите на създателите на алгоритъма са, че той осигурява 98% точност. Принципът на действие на CVSA е същият както на аналоговия уред, но се използва дискретизация на сигнала. Това позволява да се формира количествена оценка на нивото на стрес. Тя се изразява в определяне на формата на модела, като количествените оценки се базират на апроксимирана форма на шаблоните.

Резултатите на гласовите анализатори на стреса не се приемат еднозначно от научната общност. Претенциите на Института за верификация на истината се оспорват от много автори и изследователи. В [4] авторът извършва изследвания, като имитира психологичен стрес в лабораторни условия. Той модулира звуков сигнал по подобие на гласните струни и установява, че посочените в патента шаблони се получават при честотна модулация, което е в противоречие със съдържанието на патента. В резултат се прави заключението, че уредът е ненадежден. До подобни заключения достигат и други автори. В [5] са разгледани правни аспекти на този вид изследвания. Авторите не са спестили критики към всички методи, приемайки ги за шарлатанство.

3. Изследвания с метод за емпирично разлагане на сигналите (EMD)

От направеното изложение в предходната точка, може да се направят два извода:

- До момента няма техническо средство, което с висока достоверност да установява стреса в човешкия глас. Не е известно масово използване на такива технически средства от правоохранителните органи;

- Въпреки противоречивите резултати, продължават усилията за създаване на такива средства с нови методи, свързани с компютърни алгоритми и цифровата обработка на сигналите.

Физиологичният микротремор, както и човешкият глас, е нелинеен, нестационарен процес. Честотата и амплитудата на микротремора варира значително във времето. Ето защо разгледаните методи за анализ на гласовия стрес, които използват цифрова обработка на сигналите като бързото преобразуване на Фурие (FFT), нямат добра времева резолюция. Това значително намалява ефективността на отчитането на стреса. EMD е нов математически метод, който е алгоритъм за разлагане на нелинейни, нестационарни сигнали в линейна комбинация от хармонични сигнали. Това позволява да бъдат своевременно забелязани колебанията в честотата и амплитудата [6].

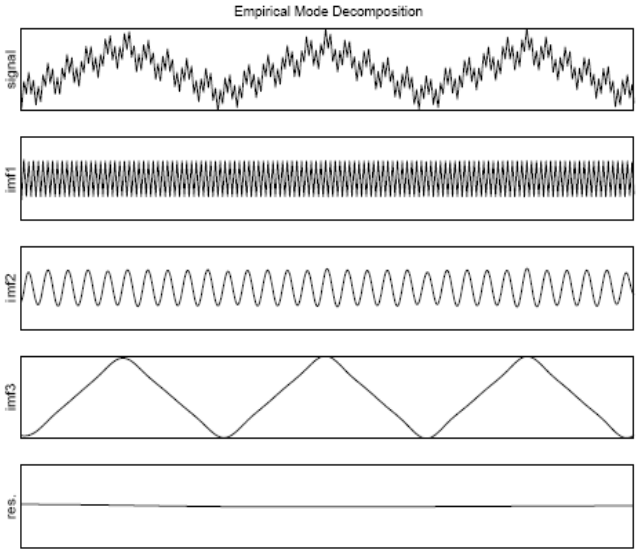
EMD работи чрез идентифициране на времеви сигнални поредици, които разкриват физическите характеристики на сигнала. Сигналът се разлага на множество стационарни сигнали, които представляват тези физични характеристики. Те са известни като Intrinsic Mode Functions (IMFs) – присъщи функции на модела. Функциите отговарят на следните критерии:

- броят на екстремумите и нулите в даден интервал трябва да бъдат или равни, или да се различават с единица;
- във всяка точка на функцията, средната стойност на обвивката, дефинирана от локалния максимум и обвивката дефинирана от локалния минимум, е равна на нула.

EMD е итеративен процес, чийто резултат е IMFs. Итерациите се прекратяват при достигане на начално зададена точност. Критерият може да бъде стандартно отклонение между два съседни резултата. Процесът се изразява в следното:

- горната и долна обвивки на сигнала се моделират с прилежащия му максимум и минимум, като се използва кубична сплайн-функция;
- извлича се средната стойност от сигнала, за да се получи нов сигнал;
- определя се дали новият сигнал е IMF, като се използват описаните по-горе критерии;
- ако новият сигнал е IMF, той се изважда от началния за итерацията сигнал. С получената разлика, процедурата започва от начало до получаване IMF.

На фиг. 2 е показано приложение на алгоритъма при разлагане на сложен сигнал на IMFs [6].



Фиг. 2. Разлагане на сложен сигнал на IMFs

С получените IMFs от EMD процеса, се извършва трансформация на Hilbert-Huang (HHT). Оригиналният сигнал се представя като линейна комбинация от реалните части на аналитичната функция. Амплитудата може да бъде представена като функция на честотата и времето. Това разпределение е известно като Хилберт-спектър.

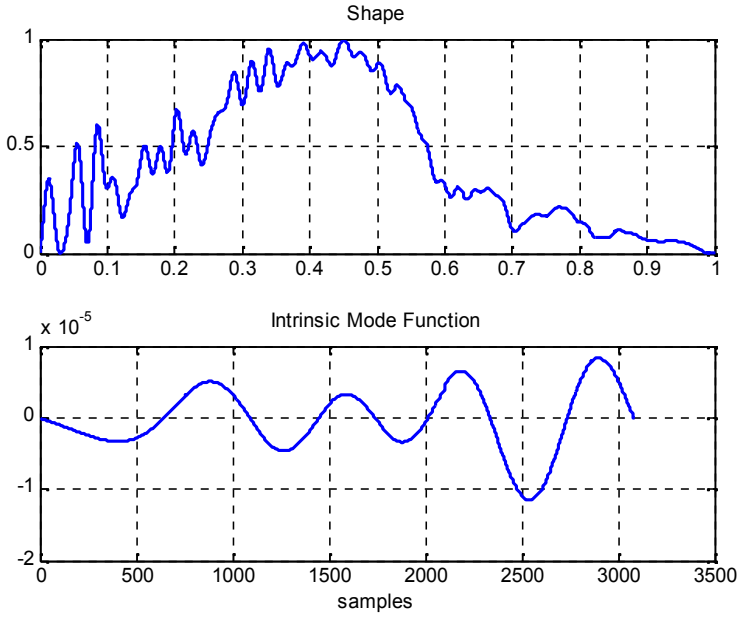
4. Експериментални изследвания

За целите на проучването в Института по психология на МВР бяха проведени 21 стандартни полиграфски теста с 16 лица-доброволци. По време на тестовете бяха направени аудио записи с професионална апаратура. Участникът записва на малък лист хартия число от 1 до 7, което полиграфистът не знае, и го поставя на стола под десния си крак (сяда върху листчето). Полиграфистът инсталира датчиците и обяснява предназначението им и съществото на теста. Тестът започва като се съобщава името и възрастта на участника. През определени интервали от време, полиграфистът задава въпрос към участника кое е числото, като последният отговаря винаги с „не". По този начин той казва неистина в единия от отговорите и това се отчита на полиграфа.

Отговорите „не" се анализират с компютърна програма на Matlab. Определя се шаблонът на думата и с цифрово интегриране се определя площта му. Успоредно с това, сигналът преминава през нискочестотен филтър с гранична честота 14Hz. Филтърът е синтезиран с метод на компресирани косинуси [7] и има параметри близки до идеален филтър.

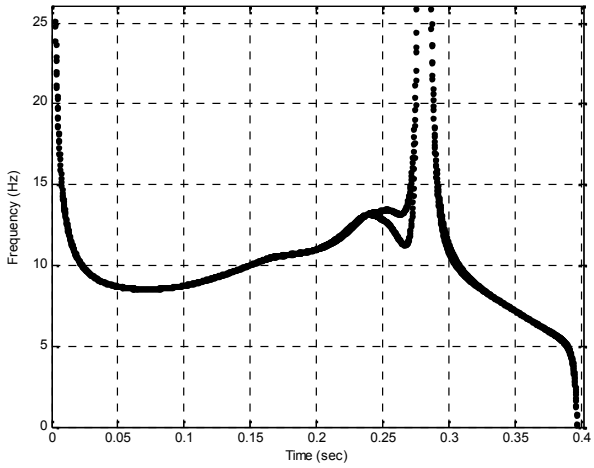
Следва прилагане на алгоритъм за определяне на присъщите функции – IMFs, и се изчислява честотата на функцията от най-висок ред. Площта на шаблона се сравнява с критериите от патентния документ на NITV.

Изходните данни съдържат информация за степента на психологичния стрес, според патентния дикумент, и честотата на IMF от най-висок ред. На фиг.3 са показани шаблонът при липса на блокиране и присъщата функция IMF с честота 10.3862Hz.



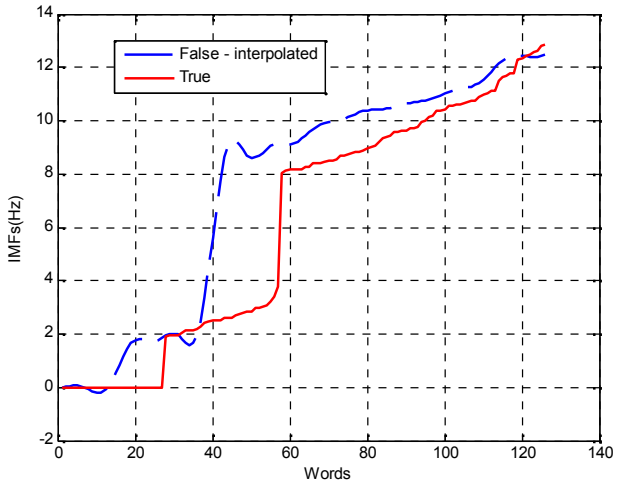
Фиг. 3. Шаблон и IMF на думата "не"

На фиг. 4 е показана Хилберт-Хуанг трансформацията на същата дума



Фиг. 4. Хилберт-Хуанг трансформация на думата “не”

На фиг. 5 са показани в графичен вид обобщени данни за честотите на IMFs, когато изследваните лица казват истина и неистина (с прекъсната линия). Данните за неистина са интерполирани, тъй като са 6 пъти по-малко на брой.



Фиг. 5. Честоти на присъщите функции IMFs на изследваните доброволци

Графиките имат приблизително еднакъв характер. Средните стойности на честотите на IMFs за неистина са

$$\overline{F}_{False} = \sum_{i=1}^{N} \frac{F_i}{N} = \frac{152.784}{21} = 7.275428\,\text{Hz},\ N = 21\,. \tag{0.1}$$

Средните стойности на честотите на IMFs за истина са

$$\overline{F}_{True} = \sum_{i=1}^{N} \frac{F_i}{N} = \frac{765.688}{126} = 6.07688\ \text{Hz},\ N = 126\,.. \tag{0.2}$$

С оглед на пълнота на изследванията, с програмите median(.) и std(.) на Matlab са изчислени статистически данни:

- медианна стойност: $F_{m\_False} = 9.1394\text{Hz}$, $F_{m\_True} = 8.2203\text{Hz}$;
- стандартно отклонение: $F_{std\_False} = 4.4047\text{Hz}$, $F_{std\_True} = 4.5216\text{Hz}$.

Визуалният оглед на формите на шаблоните и резултатите по критериите от патентния документ не потвърждава заявените претенции на Института по верификация на истината. В тази оценка важна роля играе нормирането на шаблона, по-точно определяне на продължителността на ехото в стаята (реверберацията). Необходимо е прецизно измерване с тест-сигнал с определена продължителност, каквото ще бъде направено в следващи изследвания.

От научната публикация на Олоф Липолд, се разбира, че при наличие на психологичен стрес, честотата на тремора се увеличава. От изчисленията се вижда, че средната и медианна IMFs честоти, при изговаряне на неистина, имат по-високи стойности с малко над 1Hz. От друга страна, ако честотата на психологичния тремор е над 14Hz, тогава трябва да няма IMFs, т.е. честотата да е нула. Това се наблюдава в 3 от 21 на случаите False (14.29%) и в 27 от 126 при True (21.43%), което не потвърждава тази хипотеза.

5. Изводи

В заключение може да се каже, че с предложения софтуер не може да се верифицира достоверно психологичен стрес. Вероятните фактори, които водят до този негативен резултат, са следните:

- Тестовете с изследваните лица са проведени при обстоятелства, които не провокират психо-физиологична реакция на стрес. Ако такъв тест се проведе с лице от криминалния контингент, заплашено от ефективна присъда, резултатите ще са различни.
- Броят на изследваните лица - 16, а на тестовете – 21 е твърде малък за статистически заключения.

Литература

[1]. С. Занев, Полиграфът или машината на истината. «Изток- Запад», 2002, с.32, с.33, с.63.

[2]. O. C. J. Lippold, Oscillation in the stretch reflex arc and the origin of the rhythmical, 8-12 c/s component of physiological tremor, J. Physiol. (1970), 206, pp. 359-382.

[3]. Charles Humble, Method for Quantifying Psychological Stress Levels Using Voice Pattern Samples, Patent Application Publication, № US 2005/0131692 A1.

[4]. Victor L. Cestaro, A Test of the Computer Voice Stress Analyzer (CVSA) Theory of Operation, Polygraph, 2002, 31(2), pp.72-86.

[5]. Anders Eriksson and Francisco Lacerda, Charlatanry in forensic speech science: A problem to be taken seriously, The international journal of speech, language and the law, vol. 14.2 2007, pp. 169–193.

[6]. Hai Huang, Jiaqiang Pan. Speech pitch determination based on Hilbert-Huang transform, Elsevier, Signal Processing 86 (2006), pp. 792-803.

[7]. П. С. Апостолов. Математически приближения с компресирани косинуси и техни приложения, Академично издателство «Проф. Марин Дринов», 2012.

За контакти:
доц. д.т.н. Петър Ст. Апостолов
катедра „Безжични комуникации и разпръскване”
Висше училище „Колеж по телекомуникации и пощи”
E-mail: p_apostolov@abv.bg

ИЗИСКВАНИЯ ЗА ОФОРМЯНЕ НА СТАТИИТЕ ЗА СПИСАНИЕ "КОМПЮТЪРНИ НАУКИ И ТЕХНОЛОГИИ”

I. Статиите се представят разпечатани в два екземпляра (оригинал и копие) в размер до 6 страници, формат А4 на адрес: Технически университет – Варна, ФИТА, ул. „Студентска” 1, 9010 Варна, както и в електронен вид на имейл адреси: peter.antonov@ieee.bg или jppet@abv.bg.

II. Текстът на статията трябва да включва: УВОД (поставяне на задачата), ИЗЛОЖЕНИЕ (изпълнение на задачата), ЗАКЛЮЧЕНИЕ (получени резултати), БЛАГОДАРНОСТИ към сътрудниците, които не са съавтори на ръкописа (ако има такива), ЛИТЕРАТУРА и информация за контакти, включваща: научно звание и степен, име, организация, поделение (катедра), e-mail адрес.

III. Всички математически формули трябва да са написани ясно и четливо (препоръчва се използуване на Microsoft Equation).

IV. Текстът трябва да бъде въведен във файл във формат WinWord 2000/2003 с шрифт Times New Roman. Форматирането трябва да бъде както следва:

1. Размер на листа - А4, полета: ляво - 20мм, дясно - 20мм, горно - 15мм, долно - 35мм, Header 12.5мм, Footer 12.5мм (1.25см).
2. Заглавие на български език - размер на шрифта 16, удебелен, главни букви.
3. Един празен ред - размер на шрифта 14, нормален.
4. Имена на авторите - име, инициали на презиме, фамилия, без звания и научни степени - размер на шрифта 14, нормален.
5. Два празни реда - размер на шрифта 14, нормален.
6. Резюме и ключови думи на български език, до 8 реда - размер на шрифта 11, нормален.
7. Заглавие на английски език - размер на шрифта 12, удебелен.
8. Един празен ред - размер на шрифта 11, нормален.
9. Имена на авторите на английски език - размер на шрифта 11, нормален.
10. Един празен ред - размер на шрифта 11, нормален.
11. Резюме и ключови думи на английски език, до 8 реда - размер на шрифта 11, нормален.
12. Основните раздели на статията (Увод, Изложение, Заключение, Благодарности, Литература) се форматират в едноколонен текст както следва:
    a. Наименование на раздел или на подраздел - размер на шрифта 12, удебелен, центриран, един празен ред преди наименованието и един празен ред след него - размер на шрифта 12, нормален;
    b. Текст - размер на шрифта 12, нормален, отстъп на първи ред на параграф – 10 мм; разстояние от параграф до съседните (Before и After) за целия текст – 0.
    c. Цитиране на литературен източник - номер на източника от списъка в квадратни скоби;
    d. Текстът на формулите се позиционира в средата на реда. Номерация на формулите - дясно подравнена, в кръгли скоби.
    e. Фигури - центрирани, разположение спрямо текста: “Layout: In line with text”. Номер и наименование на фигурата - размер на шрифта 11, нормален, центриран. Отстояние от съседните параграфи – 6 pt.
    f. Литература – всеки литературен източник се представя с: номер в квадратни скоби и точка, списък на авторите (първият автор започва с фамилия, останалите – с име), заглавие, издателство, град, година на издаване, страници.
    g. За контакти: научно звание и степен, име, презиме (инициали), фамилия, организация, поделение (катедра), e-mail адрес, с шрифт 11, дясно подравнено.

Образец за форматиране можете да изтеглите от адрес http://cs.tu-varna.bg/ - Списание КНТ, Spisanie_Obrazec.zip.

Този брой от списанието се отпечатва със средства от научно-изследователски проект, финансиран целево от държавния бюджет.