Information technology Applications

Aplikácie informačných technológií

1/2013

Information technology Applications / Aplikácie informačných technológií 1-2013

EUROKÓDEX
www.eurokodex.sk

ISSN 1338-6468

Paneurópska vysoká škola
Fakulta informatiky

Občianske združenie VZDELÁVANIE–VEDA–VÝSKUM
Civil Association EDUCATION–SCIENCE-RESEARCH
Некоммерческая организация ОБРАЗОВАНИЕ-НАУКА-ИССЛЕДОВАНИЕ
Andrusovova 5, 851 01 Bratislava, Slovakia
www.v-v-v.sk

Published with support from Stephan Verbovec Foundation

Information Technology Applications / Scientific Journal – 1/2013

Design:
EUROKÓDEX s.r.o. (www.eurokodex.sk)

Print:
Polygrafické centrum
Tomášikova 26
821 01 Bratislava
ba@polygrafcentrum.sk
www.polygrafcentrum.sk

Address of the Editorial Office:

Civil Association
Education-Science-Research
Andrusovova 5
851 01 Bratislava
Slovakia
vvv.esr@gmail.com

Pan European University, n.o.
Tematínska 10
851 05 Bratislava,
Slovakia
Phone: +421 2 68 203 639
martin.sperka@paneurouni.com

The journal is published four times a year.

The Magazine is Reviewed, volume 1

Subscription:
Subscription orders must be sent to the editorial Office.

The price is 20 EUR a year. It is possible to order older issues only until present supplies are exhausted (15 EUR an issue).

Published by:
Eurokodex, s.r.o. in collaboration with the Pan-European University,

Tomášiková 20, 821 02 Bratislava, Slovakia, IČO: 36 077 429

Number of copies: 120 pieces, **ISSN:** 1338-6468 (print version), **EAN** 9771338646000 07

Registration No.: EV 4528/12

Electronic version of Journal: http://www.v-v-v.sk

Delivered to the press: July 2013

Editorial Board List

Predseda redakčnej rady/ Chief editor:

doc. Ing. Martin Šperka, PhD. .....Faculty of Informatics, Paneuropean University in Bratislava

Redakčná rada / Editorial board:

prof. Mikhail A. Basarab, DSc., ...Bauman Moscow State Technical University, Moscow, Russian Federation

prof. Ing. Ivan Brezina, CSc., ...Faculty of Economic Informatics, The University of Economics in Bratislava

Dr. prof. BuchaevYakhua Gamidovich, ...Dagestan State Institute of National Economy (DGINH), Russian Federation

doc. RNDr. Andrej Ferko, PhD., ...Faculty of Mathematics Physics and Informatics, Comenius University in Bratislava

doc. Ing. Beáta Gavurová, PhD. MBA, ...Faculty of Economics, The Technical University of Košice

Ing. Michal Grell, PhD., ...Civil Association EDUCATION-SCIENCE -RESEARCH in Bratislava

doc. Ing. Ladislav Hudec, CSc., ...Faculty of Informatics and Information Technologies, Slovak University of Technology in Bratislava

prof. RNDr. Jozef Kelemen, DrSc., ...Faculty of Philosofhy and Science, Silasian University in Opava

Dr. Ing. Jaroslav Kultan, PhD., ...Faculty of Economic Informatics, The University of Economics in Bratislava

prof. V.I. Kolesnikov, ...Russian Academy of Science, Russian Federation

Ing. Eva Mihaliková, PhD., ...Faculty of Public Administration, Pavol Jozef Šafárik University in Košice

doc. RNDr. Eugen Ružický, PhD., ...Faculty of Informatics, Paneuropean University in Bratislava

doc. RNDr. Frank Schindler, PhD., ...Faculty of Informatics, Paneuropean University in Bratislava

doc. Ing. Anna Čepelová, PhD., ...Faculty of Public Administration, Pavol Jozef Šafárik University in Košice

prof. Ing. Jiří Voříšek, CSc., ...Faculcy of Informatics and Statistics, University of Economics in Prague

prof. Vladimir Zuev, ...Institute for Social and Human Knowledge, Kazan, Russian Federation

Výkonný redaktor /Managing editor

Ing. Ivana Černáková, PhD. .....Civil Association EDUCATION-SCIENCE-RESEARCH in Bratislava

Obsah / Contents

Predslov / Editorial

Vedecké články / Research papers

- **Jednoduché aplikácie IT/CI produktov v ekonomickej edukačnej sieti 4**
  Simple applications of IT/CI products in economic educational network
  Ladislav Andrášik

- **Modelový aspekt analýzy ekonomických procesov podniku a metódy riešenia 31**
  Model aspect of the analysis of companies economic processes and methods for their solutions
  Ivan Brezina, Zuzana Čičková

- **Návrh hybridného algoritmus umiestňovania dvojrozmerných útvarov na prepravnú plochu 43**
  Proposition of hybrid algorithm for solving two dimensional strip packing problem
  Ján Pittner

- **Rôzne prístupy k analýze portfólia 56**
  Various Approaches to the Portfolio Analyses
  Ivan Brezina jr.

- **Využitie algoritmu včelej kolónie pri riešení optimalizačných úloh 65**
  Application of bee colony algorithm in solving optimization problems
  Marián Šedivý

Odborné články / Survey papers

- **Skúsenosť s prechodom od LMS Moodle ku cloudovému LMS iSmart 73**
  Experience with change from LMS Moodle to cloud LMS iSmart
  Ján Papula

Informácie / Information

- **Slovak Society for Operations Research 83**

- **Cooperation of company KROS INC. and Civil Association EDUCATION–SCIENCE-RESEARCH 85**

- **Informatics 2013 – The 12th International Scientific Conference 87**

- **Instructions for authors 89**

Predslov / Editorial

Dear readers,

the scientific journal is given into your hands again, founded by the Faculty of Informatics of Pan European University in Bratislava and Civil Association EDUCATION - SCIENCE - RESEARCH in Bratislava.

Current issue is focused on the problem of quantitative methods with IT support. Among renowned authors (Brezina, Andrášik and Papula), there is also given more space to younger authors (Čičková, Brezina jr.), but also to new and upcoming authors (Pittner, Šedivý).

Single contributions are enriched by fresh and interesting topics. They are oriented to this sphere of problems:

- The analysis of using the non-linear dynamic models in qualitative scope of reasoning in economy but also in another economic disciplines (finance, management, planning disciplines) and their connection with software support,
- The use of IT for economic task's solutions using by mathematical models,
- Some criteria for intensity evaluation in solving selected problems,
- Application of innovative methods in implementing knowledge through the example of two specific e-learning systems,
- Design and implementation of a genetic algorithm focused on solving multi-dimensional backpack problem based on an evolutionary algorithm and his applied realization,
- Analysis of model approaches for solving the dynamisation of portfolio selection,
- The analysis of using Bee Colony algorithm for solving the Track Dispatching Problem and design of computer application in accordance with methodological connection of the Track Dispatching Problem analysis with the possibility of using Swarm Intelligence.

We are sure the mentioned topics will be interesting for you and could be an appropriate knowledge source in your future scientific and research work.

The journal also contains information about the activities of our partners that are written for your attention.

Bratislava, 2 July 2013

Ivan Brezina
Member of Editorial Board

Jednoduché aplikácie IT/CI produktov v ekonomickej edukačnej sieti

Simple applications of IT/CI products in economic educational network

Ladislav Andrášik

Abstract:

Thanks to rapid progress in Information Technologies, (IT) and Computational Intelligence (CI) there emerging very effective possibilities for cultivation of education processes. The author focuses attention to the process of hybridisation of economic educational networks by simple IT/CI devices and services. Conventional educational network for economics in this way, which is endowing its one by softbots, myslits and shortly speaking by experimental entities living in virtual laboratories achieving qualitatively very high level of didactics with empowering of constructive character of education. Based on several years' experience with simple IT/CI devices in social sciences and economics education the author is lays stress on such devices that can be construct in current software like Simulink, STELLA, iDMC, Vensim, SWARM and also in Excel. In this context is valid the claim "in simplicity is hidden the power". Therefore, it is no need to toil and moil after sophisticated achievements of IT/CI devices rather to consistently put into effect simple easily manageable devices, tools, methods and services. The author is brings examples of using mentioned devices for hybridisation of educational network predominantly for elementary economics in university level.

Keywords:

Constructive versus Instructive and/or Declarative approaches, Educational networks for economics, Experimentation in VirtLabs, Hybridization of network by Softbots and/or by Virtual laboratories, Storytelling's assisted by STELLA,

ACM classification: I.2, J.1, J.4, K.6

Úvod

Pri štúdiu ekonomických predmetov sa vynára pred študentmi úloha porozumieť viacerým komplexným dynamickým procesom na čo ešte väčšina z nich nie je vedomostne ani čo sa týka potrebných analytických zručností pripravená. Často sú postavení zoči-voči mentálnym modelom prevedeným do matematického formalizmu ktorým z hľadiska ich vysokej abstraktnosti nedokážu dať zrozumiteľný výklad a nevedia odhaliť ich potenciálne skryté komplexné dynamické vlastnosti. Ťažkosti vznikajú aj z toho dôvodu, že základné ekonomické predmety sú zaradené spravidla do prvých dvoch semestrov, takže sú paralelné s výučbou matematiky. Takto sa stáva,

že niektoré z mentálnych modelov si vyžadujú vedomosti a zručnosti z matematiky už skôr než prídu na rad v matematických predmetoch.

Našťastie pokrok v IT, v programovaní a existencia konkrétnych produktov komputačnej inteligencie (CI – computational intelligence) prichádzajú na pomoc vo viacerých takých prípadoch. Pozoruhodné je najmä to, že už relatívne veľmi jednoduché produkty, resp. služby tohto druhu vedia výrazne pomôcť hlbšiemu porozumeniu komplexnej ekonomickej dynamike. Preto zaradenie takýchto produktov do ekonomickej edukačnej siete predstavuje významný pedagogický a didaktický pokrok a znamená vyššiu efektivitu pri získavaní vedomostí aj vďaka tomu, že má pozitívny psychologický vplyv na študentov v tom, že sa odbúravajú averzie voči abstraktným konštrukciám. Ďalším pozitívnym prínosom je to, že študenti majú možnosť okrem deklaratívnych a inštruktívnych postupov používať konštruktívne metódy a nástroje pre virtuálne experimentovanie, čím sa čiastočne odstraňuje nedostatok ekonómie spočívajúci v nemožnosti prirodzeného experimentovania v objektívnej sociálno-ekonomickej realite.

Medzi softvérmi, ktoré sa nám dlhodobo osvedčili v edukačnom procese na STU[1)] a v súčasnosti sa ich snažíme zavádzať aj na EUBA[2)] patria Simulink MATLABU, STELLA, iDMC a samozrejme nevynímajúc osvedčený bunkový procesor Excel. Konštruktívne myslenie výrazne podporuje najmä Simulink a STELLA, resp. Vensim a podobné softvéry, napríklad SWARM. V tomto článku zameriavame pozornosť predovšetkým na využívanie jednoduchých a ľahko ovládateľných didaktických postupov podporovaných zmienenými softvérmi, osobitne so softvérom STELLA, iDMC a samozrejme aj Excel, ktorý je medzi ekonómami azda najobľúbenejší pomocník. Pomocou nich sa snažíme hybridizovať konvenčné edukačné siete pre ekonómov.

1. Hybridizácia konvenčnej edukačnej siete produktmi typu VirtLab

S nástupom IT a osobitne Internetovej revolúcie vznikli úplne nové a vysoko efektívne možnosti pre tvorbu edukačných sietí. Týka sa to tak inštitucionalizovaných ako aj spontánnych sietí a pre ekonómiu je to veľká výhoda z viacerých dôvodov najmä preto, že nemá možnosť experimentovať v objektívnej realite. Nanešťastie v našich podmienkach sa na vysokých školách pri výučbe ekonomických predmetov tieto produkty a služby sa nevyužívajú v takej miere ako by si to zaslúžili, ale najmä v miere, ktorá je nevyhnutná na hlboké porozumenie komplexných dynamických javov v hospodárstvach v podmienkach globálnej vedomostnej spoločnosti. Iniciatívnejší študenti spontánne hľadajú na Internete možnosti dozvedieť sa viac o úlohách, ktoré ich hlbšie zaujímajú. Je to pozitívny jav ale bolo by ešte lepšie, keby mali priamu možnosť získavať hlbšie poznatky na inštitucionálnych edukačných sieťach vybavených špecifickými IT/CI produktmi. V 6. RP EK sa začala širšie realizovať stratégia NoE (Network of Excellence). V rámci nej sa autor tohto článku zúčastnil ako koordinátor za SR na projekte NoE DIME (Dynamics of Institutions and Market in Europe), ktorý bol ako otvorená sieť oficiálne zavŕšený v januári 2013. Stratégia NoE, ktorá aj v podobe DIME ďalej pokračuje je veľmi vhodnou platformou na tvorbu hybridizovaných edukačných sietí pre ekonómiu aj ďalšie ekonomické odbory, prípadne aj ďalšie *spoločenské a kognitívne vedy*. Žiaľ v našich podmienkach sme v tejto oblasti nezaznamenali prílišný entuziazmus najmä u učiteľov ekonómie, kde by to bolo azda najviac potrebné. Dosť výrazne a dlhodobo pretrváva zotrvačnosť v používaní konvenčných pedagogických a didaktických prostriedkov, nástrojov a metód, hoci práve ekonómia sa dostáva pod silnú paľbu kritiky pre svoju neschopnosť vidieť hlbšie procesy v komplexnej dynamike hospodárskeho diania v podmienkach globálnej vedomostnej spoločnosti. Kritika sa výrazne zostrila v ostatných rokoch pod vplyvom finančných machinácií v USA, ktorých negatívne dôsledky sa rýchlo preniesli

1) Slovenská technická univerzita.

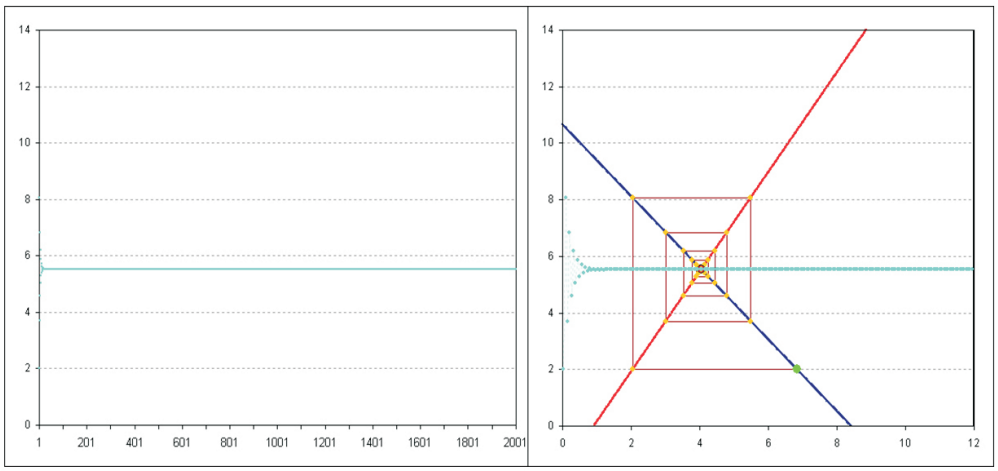
2) Ekonomická univerzita v Bratislave.

do svetovej ekonomiky a nie celkom správne či presne sa označujú pojmom kríza. Ide o to, že v ekonomickej terminológii sa kríza používa na označenie jednej z *fáz ekonomického cyklu*. Predmetný proces má skôr *charakter katastrofy* v zmysle *matematickej teórie katastrof*. Žiaľ aj v hovorovom zmysle tohto pojmu ide viac o katastrofu než o krízu. Kríza má v dohľadnom horizonte východisko, pretože je to cyklus, ale katastrofa, v tomto prípade tzv. špičková katastrofa (cusp catastrophe) v dôsledku toho, že obsahuje veľkú hysteréznu slučku má silnú tendenciu uviaznuť v nej ako v pasci na veľmi dlhú dobu. Metaforicky by sa to dalo vztiahnuť aj na katastrofické deje v Eurozóne, ktoré sa sotva dajú riešiť konvenčnými nástrojmi. Práve možnosť nasadiť postupy a nástroje IT/CI na riešenie podobných úloh sľubuje vniesť viac svetla do takých zvláštnych veľmi komplexných dynamických procesov, pravda to nie je ambíciou tohto článku, ktoré je zameraný úzko na kultiváciu edukačných procesov v ekonómii s využitím IT/CI. Skúsenosti, ktoré sme získali po desaťročiach aktívnej výskumnej a pedagogickej práce v tejto oblasti nás utvrdili v tom, že pre rané štádiá získavania ekonomických vedomostí a zručností na vysokých školách sa osvedčila práca s *mentálnymi modelmi budovanými odhora nadol*, čo je prípad takmer všetkých mentálnych modelov v ekonómii následne pretransformovaných do matematických formalizmov. Pokročilejšie aktivity, resp. vedecko-výskumné či vedecké kvalifikačné aktivity môžu lepšie využiť *modelovanie odspodu nahor*. V prípade tých prvých ide vlastne o analýzu mentálnych modelov menovitého tvorcu a nie o analýzu ekonomickej reality, tak ako napríklad *ekonometria* takisto neanalyzuje objektívnu sociálno-ekonomickú realitu priamo lež určitý súbor minulých čísel vyňatých zo štatistickej databázy vyťaženej z objektívnej reality, ktorá je pôvodným tvorcom udalostí a entít, ktoré reflektuje štatistika. Modely budované odspodu nahor na *základe princípu minima* predpokladov tvorcu *evolvujú virtuálny svet* a až ten sa analyzuje a porovnáva s dejmi v objektívnej realite.

2. Obyčajné IT/CI produkty na hybridizáciu ekonomickej edukačnej siete

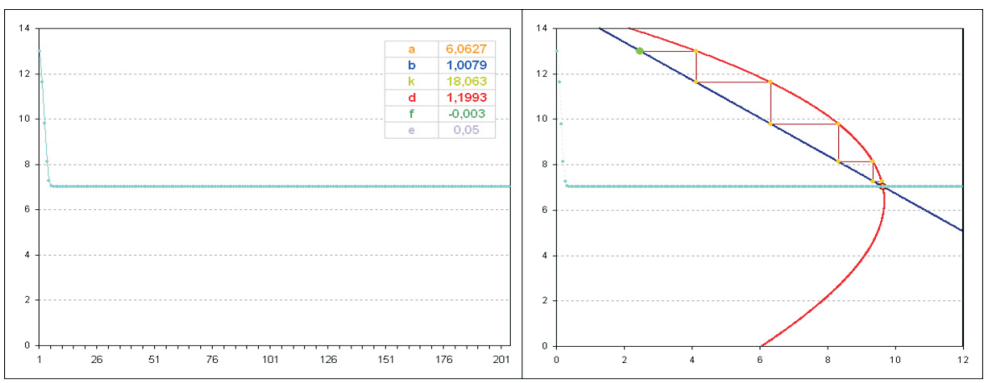
Na hybridizáciu konvenčných edukačných sietí sa môžu úspešne použiť aj veľmi jednoduché produkty, ako sú napríklad *applety*. Viaceré z nich sa celkom spontánne objavujú na Internete a tak sa aj používajú, teda z pohľadu vzdelávacích inštitúcií neintencionálne, ba niektoré vysoké školy direktívne zakazujú niektoré produkty podobného druhu používať. Ani autor tohto článku ich nepreferuje pre vkladanie do inštitucionálnych edukačných sietí avšak z iného dôvodu, totiž vzhľadom na ich fixovanú konštrukciu študenti s nimi nemôžu konštruktívne pracovať, ale iba ťažiť z nich to, čo do nich vložil ich autor. Reálny prínos v zmysle hybridizácie vidíme v možnosti konštruktívneho dialógu študenta s produktom charakteru IT/CI. Tento účel plnia už aj najjednoduchšie mentálne modely vytvárané napríklad v bunkovom procesore Excel. Na konštrukciu jednoduchých experimentálnych nástrojov nepotrebuje mať študent programátorské zručnosti ba ani len vo *Visual Basicu*. Ukážeme to na bežnom príklade modelu tzv. *pavučinového trhu*, ktorý je štandardným prvkom predmetu ekonómia. V učebniciach sa zvyčajne uvádza kresba pavučiny na dvoch spojitých priamkach, pritom však trhové procesy sú komplexnejšie a prinajmenšom ich treba vyjadriť krivkami a nie priamkami. V tom prípade však potenciálna dynamika trhu môže byť natoľko komplexná, že bežný študent nie je vybavený na to, aby si také situácie vedel zmysluplne predstaviť. Nižšie uvádzame niekoľko výstupov zo simulačných experimentov realizovaných v Excele, pretože inak sa to pri tlači na papier vlastne ani nedá, práve to ilustruje jeden z veľkých nedostatkov konvenčného prístupu v edukácii.

Obr. 1 *Typická pavučina predstavuje konvergenciu k rovnováhe trhu*

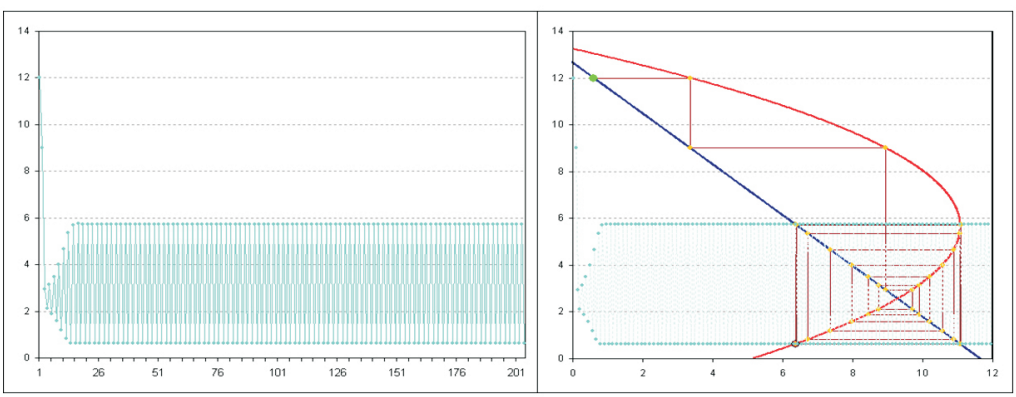


Na snímke obr. 1 je ako prvý príklad uvedený s priamkou dopytu a priamkou ponuky. Študent má možnosť meniť hodnoty štyroch parametrov priamok a zadať východiskovú cenu a sledovať dôsledky týchto zmien na diskrétnu dynamiku daného virtuálneho trhu. Na snímke obr. 2 je výsledok experimentu realizovaného v hornej časti *sigmoidnej krivky*, ktorá vznikla pridaním dvoch nelineárnych členov k pôvodnej lineárnej rovnici grafu ponuky.

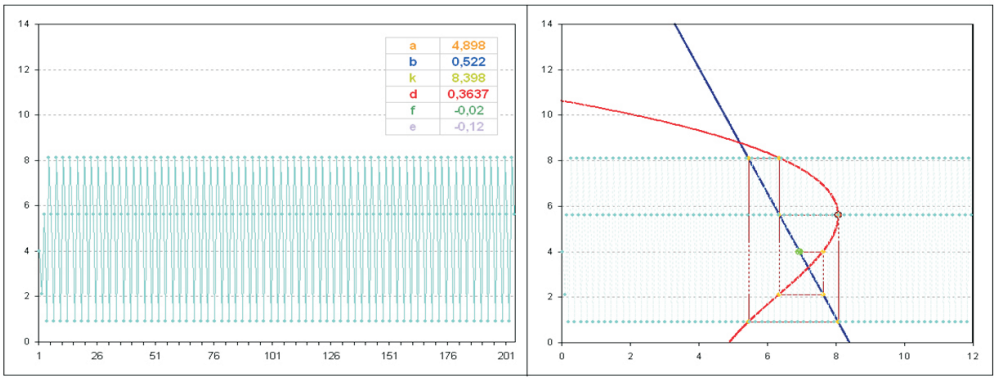
Obr. 2 *Konvergencia k trhovej rovnováhy pri dozadu ohnutej krivke ponuky*



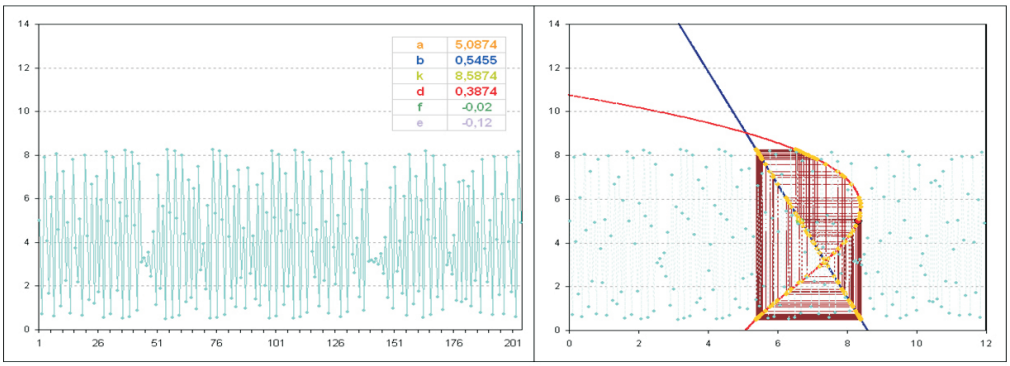
Obr. 3 *Emergencia 2-periódového cyklu*



Obr. 4: *3-periódový cyklus v hornej polohe krivky ponuky*

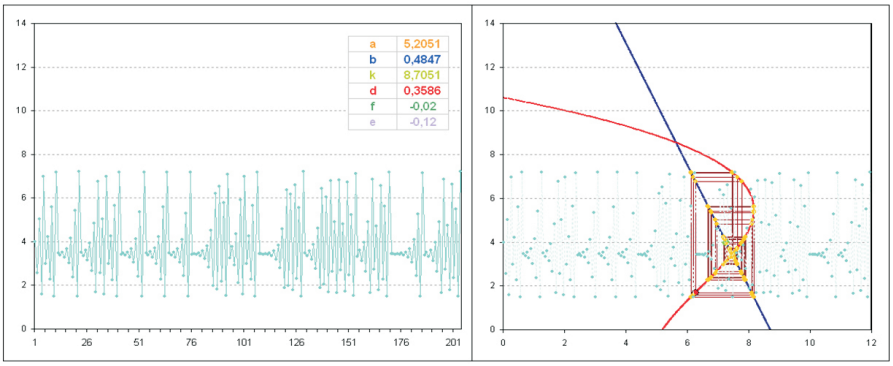


Obr. 5: *Deterministický chaos v hornej polohe krivky ponuky*

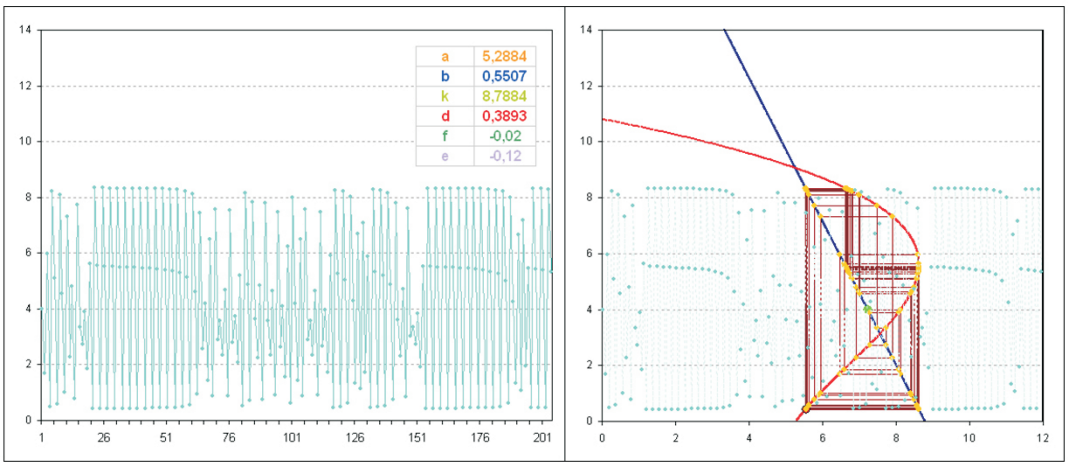


Túto aj iné podobné operácie môže bez ťažkostí urobiť každý študent na rozdiel od appletov na Internete, kde sa to nedá pretože sú autorsky zafixované. Nie je na to ani zmysluplný dôvod, pretože už jednoduchšie činnosti v Excele umožňujú efektívnejšie a efektnejšie riešenia. To dokumentujeme nasledovnou sériou snímok z experimentov v Excele. Zavedením nelineárnych členov vznikla v hornej časti kubickej krivky tzv. *backward bending supply curve*. V tejto polohe krivky môže študent experimentovať tentoraz až s hodnotou šiestich parametrov a tiež so začiatočnou cenou. Keďže vo väčšine prípadov dozadu stočenej krivky ponuky sa má na mysli *trh pracovných síl*, lepšie by sa hodilo hovoriť o mzdovej sadzbe. To čo v tlačenej podobe zaberá veľa miesta, študent pri experimentovaní vidí priamo v pohyboch (posunoch) čiar a v zmenách ich zakrivenia a najmä vidí pohyb stavového bodu.

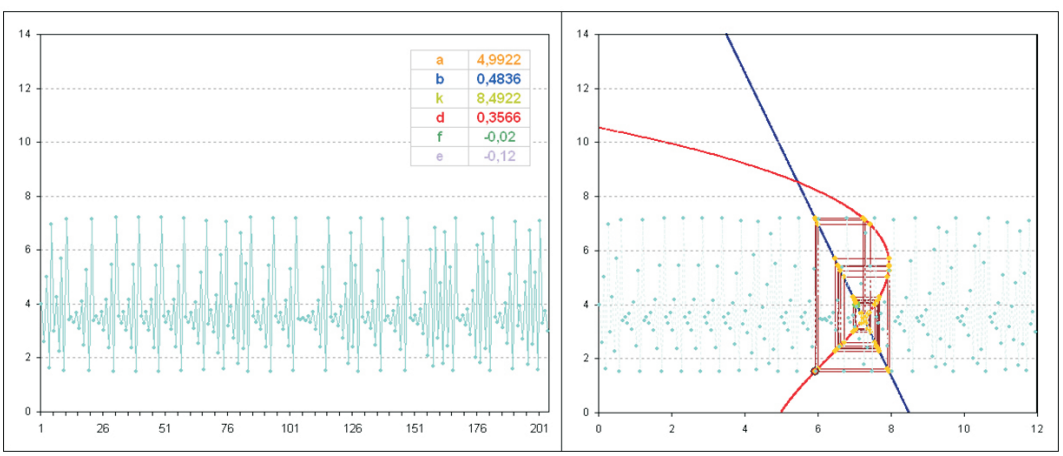
Obr. 6: *Deterministický chaos – rezonancie rovnovážneho bodu*



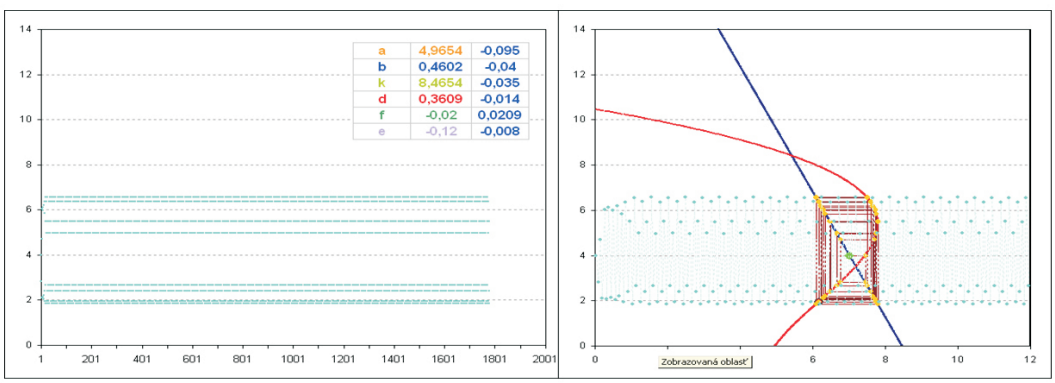
Obr. 7: *Kombinácia deterministického chaosu s 3-periódovou rezonanciou*



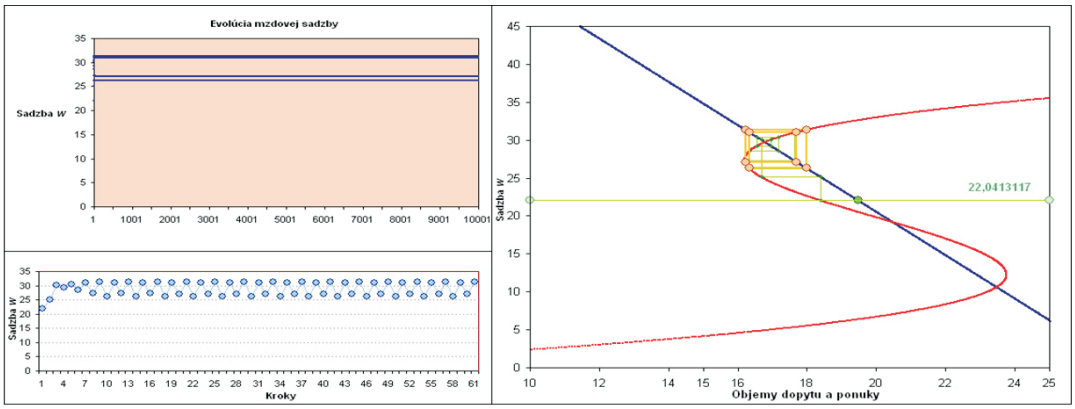
Obr. 8: *Hustá chaotická frekvencia v hornej časti krivky ponuky*



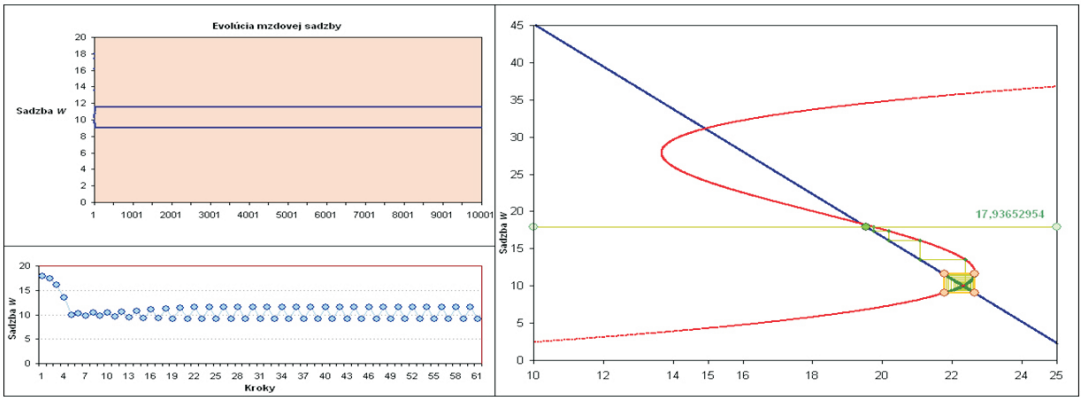
Obr. 9: *Cyklus s ôsmimi periódami*



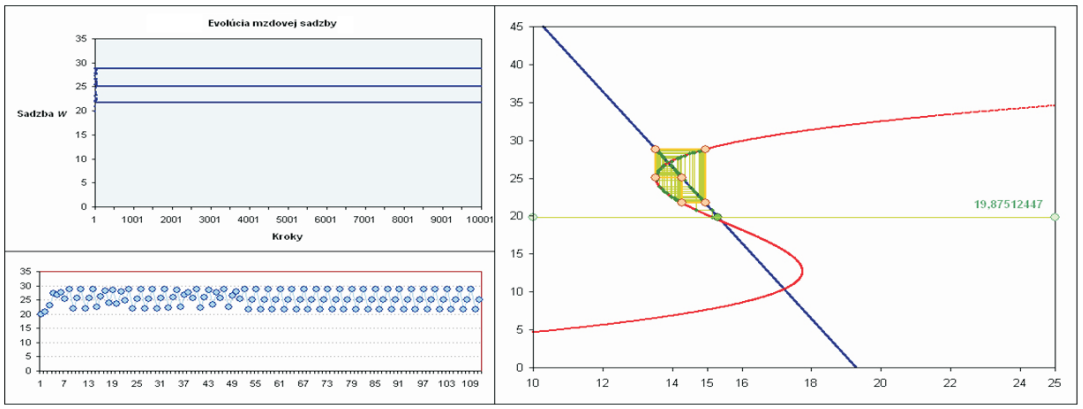
Obr. 10: *Snímka experimentu s úplnou sigmoidnou krivkou v prvom kvadrante*



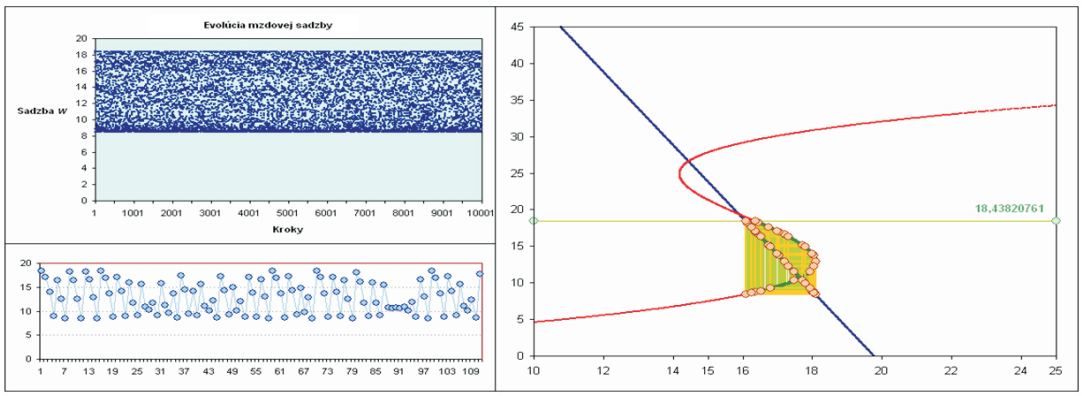
Obr. 11: *2-periódový cyklus v spodnej časti sigmoidnej krivky*



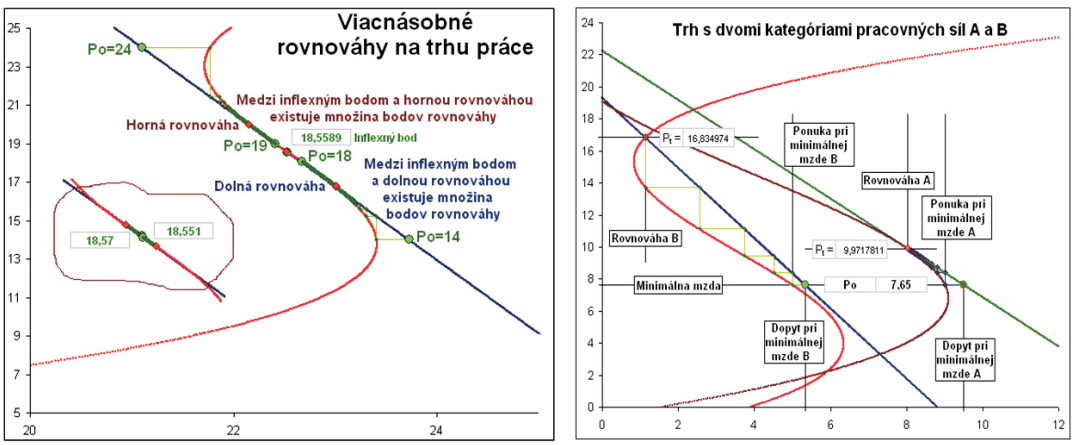
Obr. 12: *Po začiatočnom chaose sa rýchlo dosiahla 3-periódová sukcesia*



Obr. 13: *Deterministický chaos v spodnej časti sigmoidnej krivky*



Obr. 14 *Situácie získané experimentom v Excele dodatočne vybavené verbálnymi legendami*



Tak ako sme poznamenali už skôr *mentálne modely budované zhora nadol* sa pomocou experimentovania vo virtuálnom laboratóriu podrobujú analýze, ako subjektívny produkt kohosi, v danom prípade konkrétne M. Ezekiela a ďalších ekonómov. To je pre študenta inšpiratívne v tom, že aj on môže konštruovať svoje vlastné mentálne modely a podrobovať ich kritickej analýze s asistenciou produktov a služieb IT/CI. V ďalšom štádiu štúdia potom, keď nadobudne patričné matematické vedomosti a zručnosti pristúpi k analýze na vyššej vedeckej úrovni, čo je prednosťou matematiky. Matematika, kybernetika, synergetika a teória dynamických systémov je najvyššia úroveň kultivácie vedeckého myslenia aj v ekonómii a nie je to samoúčel akýchsi matematických a počítačových čudákov, ako si to žiaľ viacerí ekonómi nesprávne vykladajú.

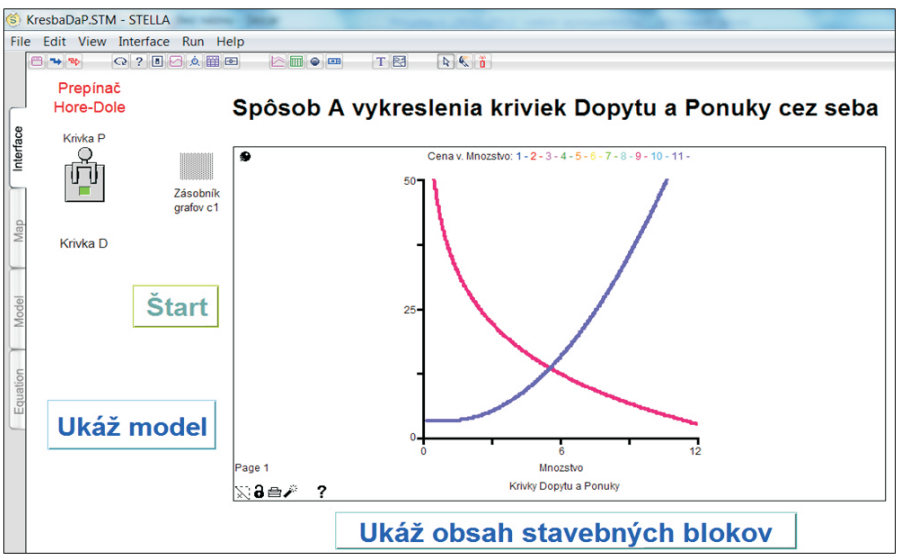
Pavučinové modely opisujú procesy na *virtuálnom trhu* veľmi názorne. Presvedčili sme sa o tom v prípade trhu s priamkami dopytu a ponuky, ale aj v prípade trhu pracovných síl, keď krivka ponuky pracovných síl bola spätne zakrivená (*backward bending curve*, resp. *backward sloped curve*) ako aj v modeli so *sigmoidnou krivkou ponuky*, ktoré sme predstavili v jednoduchom laboratóriu vytvorenom v Excele. V skôr predstavených prípadoch sa však nebrali do úvahy očakávania trhových subjektov týkajúcich sa vývoja ceny. Použitý pavučinový model opisoval časovú rovnováhu (nerovnováhu) na jednotlivom trhu keď sa pri danej cene ponuka oneskorovala o jeden časový dielec. Tento model zaviedol v 30-tych rokoch už skôr spomínaný Mordechai Ezekiel (1938) [11] a veľmi sa osvedčil aj v učebniciach ekonómie – stal sa vlastne učebnicovou vzorovou úlohou pri výklade diskrétnej ekonomickej dynamiky (matematicky ide

o topologické projekcie uskutočňované paralelne na dvoch čiarach, čo sa však v učebniciach neuvádza). Presnejšie povedané, spravidla sa v takých prípadoch predpokladá, že keď subjekty (hráči) na strane ponuky postupujú podľa modusu „naivných očakávaní" a čiary dopytu a ponuky sú v matematickom vyjadrení monotónne, môžu sa vyskytnúť len tri prípady dynamiky a to:

- konvergencia k stabilnému rovnovážnemu bodu (atraktorom systému je teda jedinečný bod, t. j. equilibrium ...*E*),
- konvergencia k postupnosti dvoch úrovní cenových bodov (Max a Min sa striedajú pravidelne), ide teda o cyklus s dvomi periódami, a napokon tretí prípad,
- explozívne vzďaľovanie sa od začiatočnej ceny, t. j. amplitúda medzi horným a dolným bodom sa ustavične v každom nasledujúcom kroku zväčšuje.

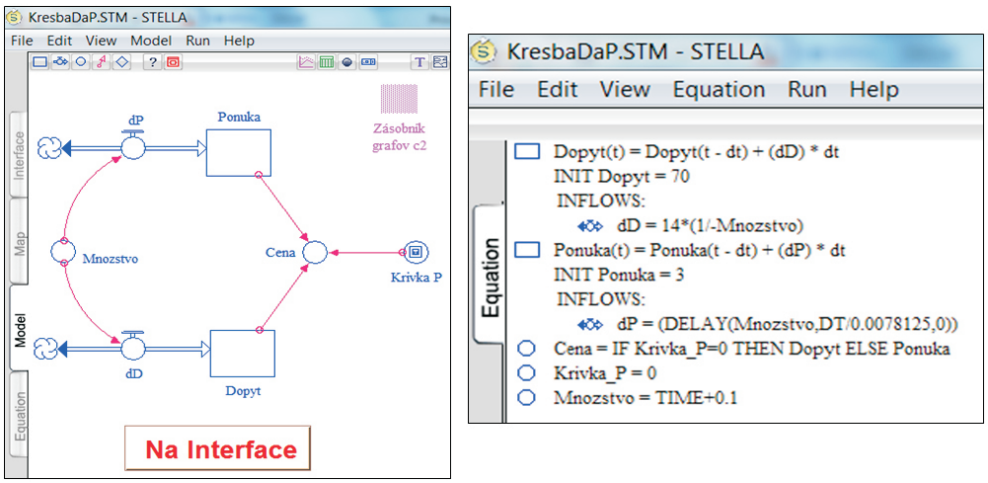
Isteže, nemalo by sa pustiť zo zreteľa, že aj prvý prípad je vlastne cyklus (v prípade totožnosti začiatočnej ceny s rovnovážnou cenou je rozpätie medzi Max a Min nulové) v ostatných prípadoch ide o tlmenú osciláciu, ktorá končí v *E*. Väčší dôraz sme však položili na nemonotónnu krivku ponuky, pretože ako sme videli v tých prípadoch je dynamika komplexnejšia a môže sa objaviť aj deterministický chaos. Trh práce, resp. *pracovnej sily*[3)] sa vyznačuje niektorými zaujímavými a zároveň aj veľmi dôležitými špecifikami. Najcharakteristickejšou črtou tohto trhu je *dozadu sa odkláňajúca krivka ponuky práce*, t. j. *backward bending*, či *backward sloping supply curve*, o ktorej sme referovali už skôr. Pre porovnanie s výsledkami dosiahnutými v Excele uvádzame príklady s modelom trhu aj v STELLE a iDMC.

Obr. 15: *Pavučinový model trhu realizovaný v STELLE: nemonotónne krivky*

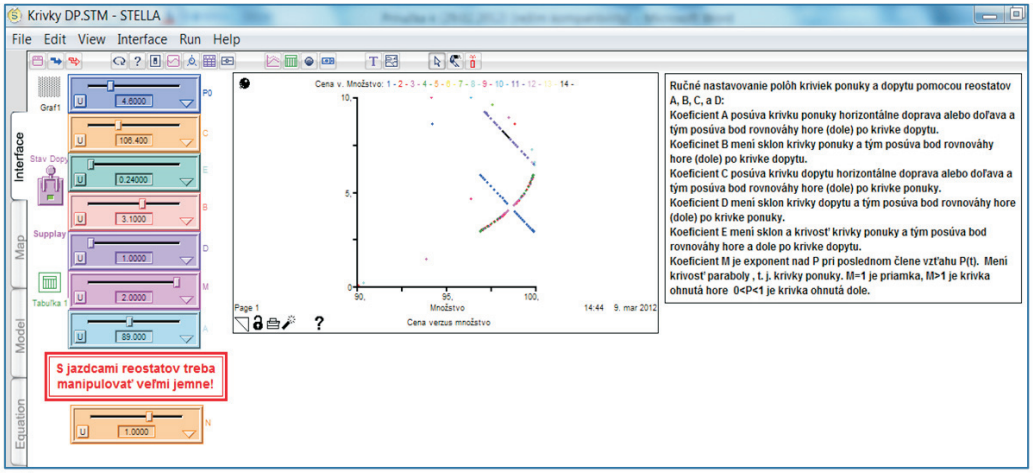


3) Dôsledne vzaté, na trhu nevystupuje práca, ale pracovná sila, t. j. *potenciálna práca*. To, čo sa implicitne predpokladá, čiže *práca* sa udeje až v *pracovnom procese*, kde sa potenciál premieňa na reálny výkon. Na trhu sa teda obchoduje s potenciálom a práca je výsledok pracovného procesu, čo je veľký rozdiel. Pri kúpe pracovnej sily sa výrobcovia snažia tlačiť ceny dole a pracovníci hrajú mimikry, či rituály o svojich vysokých schopnostiach, aby dosiahli čo najvyššiu cenu. *V pracovnom procese* je to presne naopak: manažér chce z nakúpenej pracovnej sily (pri zminimalizovanej cene) vyžmýkať čo *najviac výsledkov*, t. j. práce a pracovník sa snaží svoju pracovnú silu (ktorú predal za maximálnu cenu) čo najviac šetriť, t. j. odovzdať čo *najmenej práce* (Jeremiáš Bentham – *kalkul slasti a strasti*). Preto *až kvalita manažmentu rozhodne o skutočnej hodnote práce a nie trh pracovnej sily*. Trh rieši len vzťah medzi kvantitou ponúkanej pracovnej sily a kvantitou dopytovanej pracovnej sily, pomocou pohybu ceny, t. j. mzdovej sadzby (*wage rate*). Naozajstnú a konečnú hodnotu práce určí až trh hotových výrobkov, t. j. až keď sa na trhu realizujú ako tovary. Ak sa výrobok nestane tovarom práca bola premrhaná a cena zaplatená za pracovnú silu je čistá strata navyše umocnená tým, že sa premrhal aj materiál, palivá a energia, a amortizácia strojov a zariadení (HIM, NIM a DIM) vynaložené na vyhotovenie produktu.

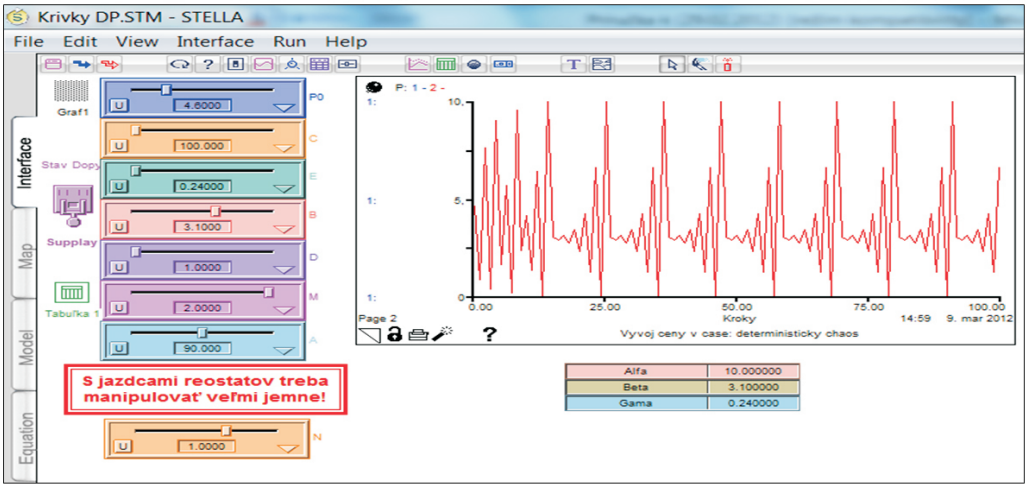
Obr. 16: *Pavučinový model trhu nemonotónnych kriviek: bloková a principiálna schéma*



Obr. 17: *Pavučinový model trhu pracovných síl (zaklonená krivka ponuky)*

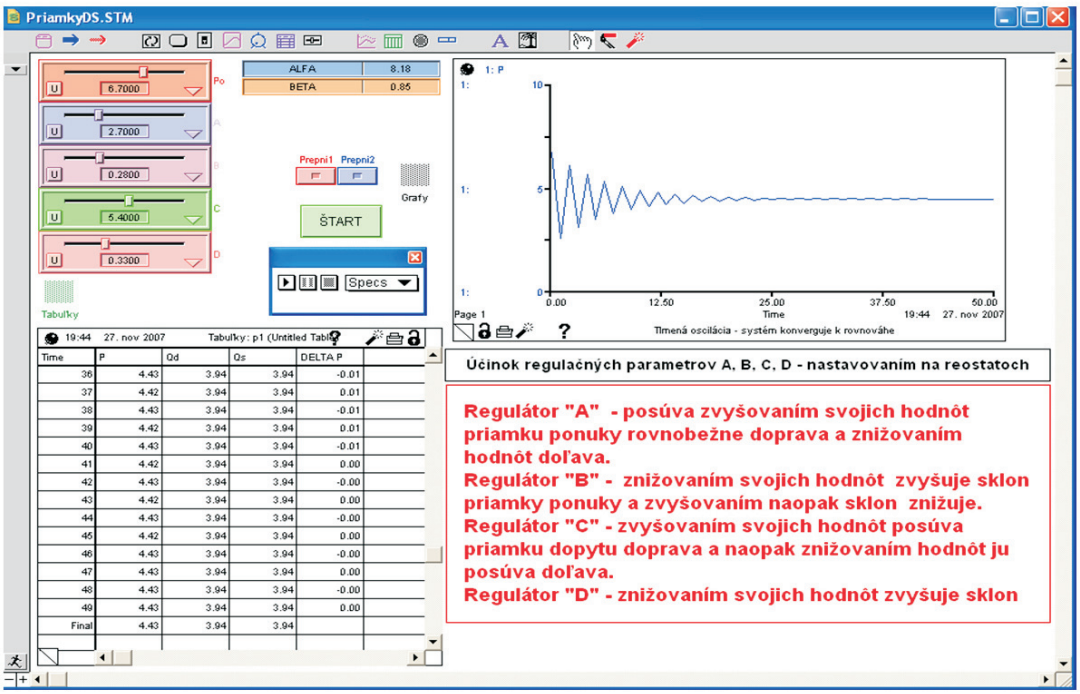


Obr. 18: *Pavučinový model trhu pracovných síl: kroková evolúcia viedla k chaosu*



Najčastejšie sa tento tvar krivky vysvetľuje tým, že pri vysokých cenách práce ponúkaná práca nemá dostatočnú kvalitu, resp. nie je primerane kvalifikovaná a keďže niektorí pracovníci chcú túto cenu dosiahnuť, musia si zvýšiť kvalifikáciu. Títo potenciálni pracovníci však v dôsledku toho nie sú na trhu, musí sa čakať kým potrebnú kvalifikáciu dosiahnu a preto v takýchto vyšších cenových vrstvách sa krivka stáča dozadu, t. j. doľava – takže v dôsledku toho *pri raste ceny v tejto vrstve ponuka práce klesá*. Po dosiahnutí vyššej kvalifikácie prichádzajú s časovým odstupom naspäť na trh pracovníci, ktorí sa vzdelávali čo je typická situácia v ére globálnej vedomostne spoločnosti. Keď študenti použijú na rovnakú úlohu rutiny, ktoré sú predprogramované v prostredí iDMC, napríklad rutiny *Bifurcation*, môžu získať celé spektrum kvalitatívnych udalostí vo zvolenom intervale konkrétneho parametra. Dokonca môžu použiť experiment s dvomi parametrami voči sebe a tak dostať kvalitatívne diverzifikovanú takmer spojitú plochu ktorá prináša informácie o periodicite patriacej ku každému bodu na 2D ploche, t. j. súradnice začiatočného bodu potenciálnej trajektórie stavového (resp. fázového) bodu na ploche $\zeta^{parA} \times \zeta^{parB}$.

Obr. 19: *Konvergencia k rovnováhe na trhu*



Inak ako v skôr demonštrovaných prípadoch sa mentálny model bude správať, ak sa doň zavedú *subjektívne očakávania aktérov na trhu*. To demonštrujeme v nasledovnej časti článku, ktorý opierame o idey Carsa Hommesa [15]. V tomto prípade sme sa opierali o asistenciu softvéru iDMC[4]. Tentoraz ešte pred vytvorením laboratória v iDMC použijeme aj trochu

4) Tento softvérový nástroj umožňuje efektívne a s vysokou názornosťou skúmať stabilitu ustálených stavov a emergenciu *periodických orbít*, vrátane *deterministického chaosu* (napríklad pomocou pavučinového znázornenia), na základe štandardných numerických simulačných techník, ako sú časové rady, vykresľovanie fázových a *bifurkačných diagramov* (resp. portrétov a máp), *Ljapunovovych exponentov* a 2D L-máp, vykresľovanie bazénov atraktorov, resp. repulzorov, vytváranie mnohovariantných (variovanie s jedným parametrom pri konštantnosti ostatných) grafov a podobne. Pre predstavenie možností tohto systému sme zámerne vybrali pomerne ľahkú ekonomickú úlohu *adaptívnej cenovej tvorby producenta*, resp. podnikateľa. Na takomto modeli totiž môžeme predviesť takmer všetky možnosti, ktoré ponúkajú u nás dostupné softvéry pre kvalitatívne experimentovanie so simulačnými modelmi.

matematického vyjadrovania. V prípade *naivných očakávaní* producent jednoducho predpokladá, že sa cenový trend zachová aj pre nasledovné obdobie, a preto formuje svoje očakávania v súlade s trendom. Zopakujeme najprv skôr demonštrovanú úlohu: producent adjustuje svoj výrobný proces na základe ceny na trhu existujúcej v mínus prvom období. Túto situáciu (ide o *konvenčnú koncepciu pavučiny* na trhu) môžeme na pripomenutie matematického pozadia mentálneho modelu zapísať nasledujúcimi jednoduchými funkciami

$$D_{(p_t)} = a - dp_t . \qquad (Dopyt) \qquad (1)$$

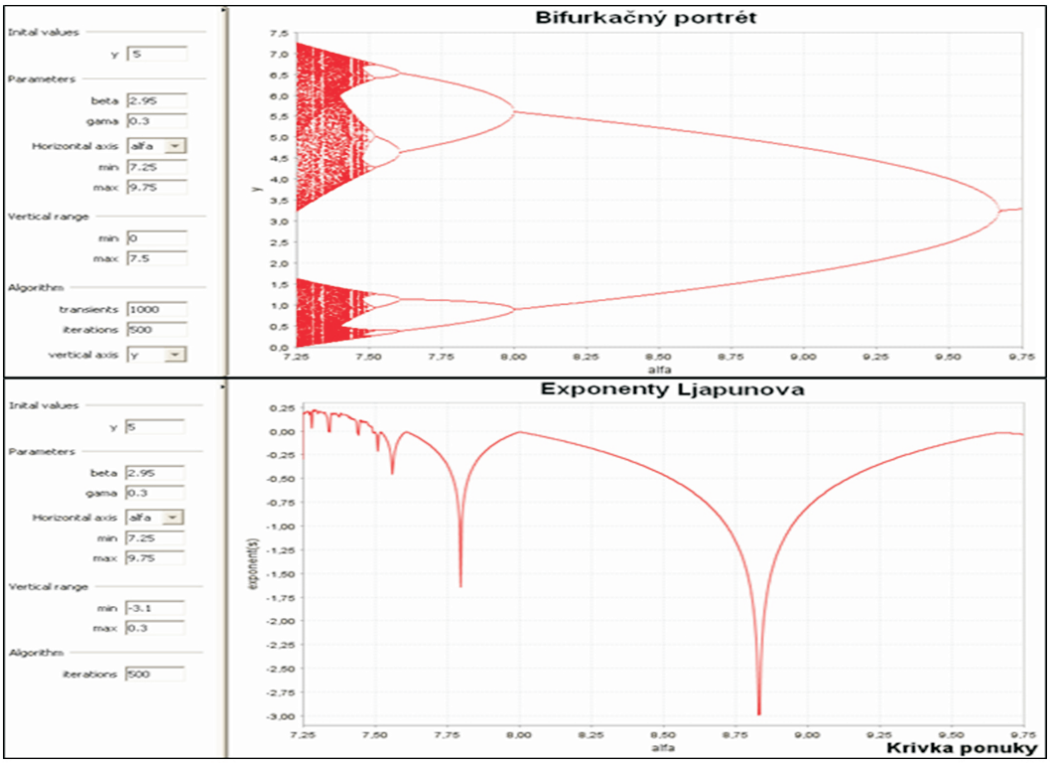
$$S_{(p_t^e)} = c + bp_{t-1} . \qquad (Ponuka) \qquad (2)$$

$$D_{(p_t)} = D_{(p_t^e)} . \qquad (Čistenie\ trhu) \qquad (3)$$

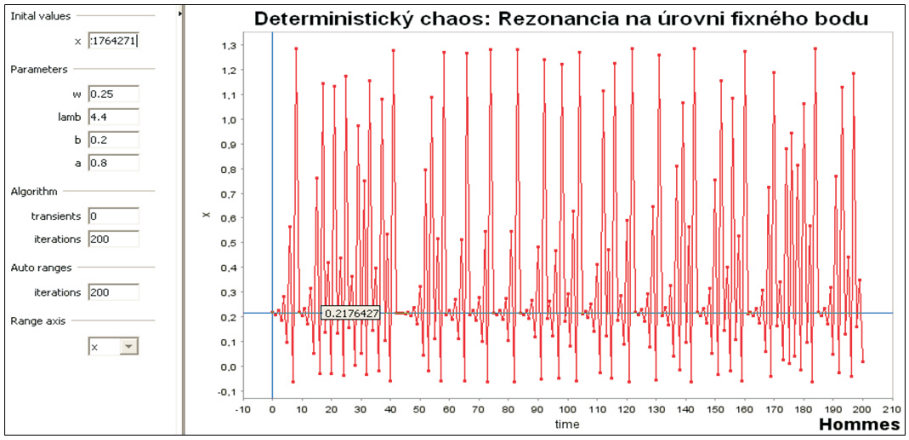
$$p_t^e = \frac{a-c}{d} - \left(1\frac{b}{d}\right)p_{t-1} + p_{t-1} . \qquad (Evolúcia\ ceny) \qquad (4)$$

Vzťahy podobné ako (1) až (4) sme vlastne použili v rôznych podobách pri predchádzajúcich simulačných experimentoch. Rovnica očakávanej ceny alternatívne zapísaná ako je to v (4) sa bežne nachádza v učebniciach, ale tento zápis sme si požičali od Hommesa [15]. Snímky obr. 21 až obr. 27 vznikli pri experimentovaní s 1D model C. Hommesa v rutinách iDMC.

Obr. 20: *Porovnanie bifurkačných hodnôt s exponentmi Ljapunova v iDMC*

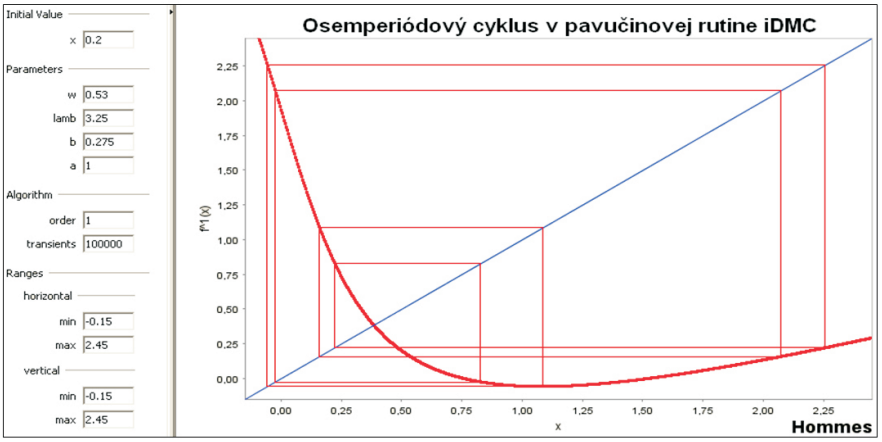


Obr. 21: *Model Hommesa v 1D prevedení – časovo-kroková trajektória v iDMC*

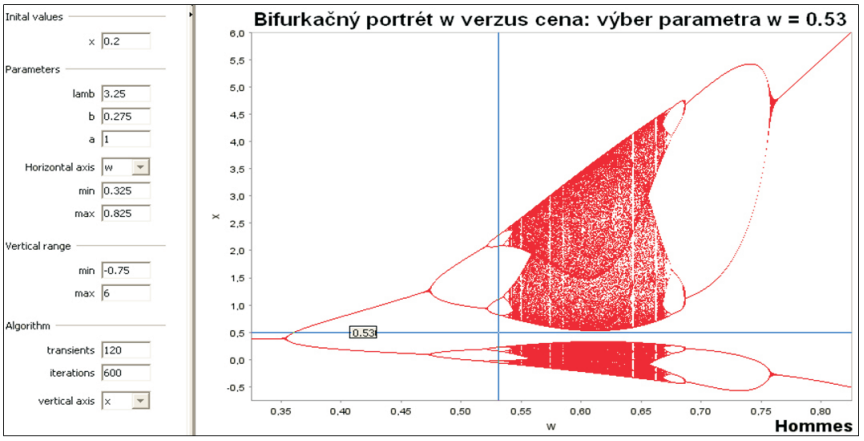


Stále preferujeme veľmi jednoduché prípady mentálnych modelov, aby bolo zrejmé, že už aj na takej úrovni môže nastať pre študentov ťažkosti pri imaginácii komplexnej dynamiky, v ktorých však môžu veľmi dobre pomáhať prostriedky, služby a metódy IT/CI.

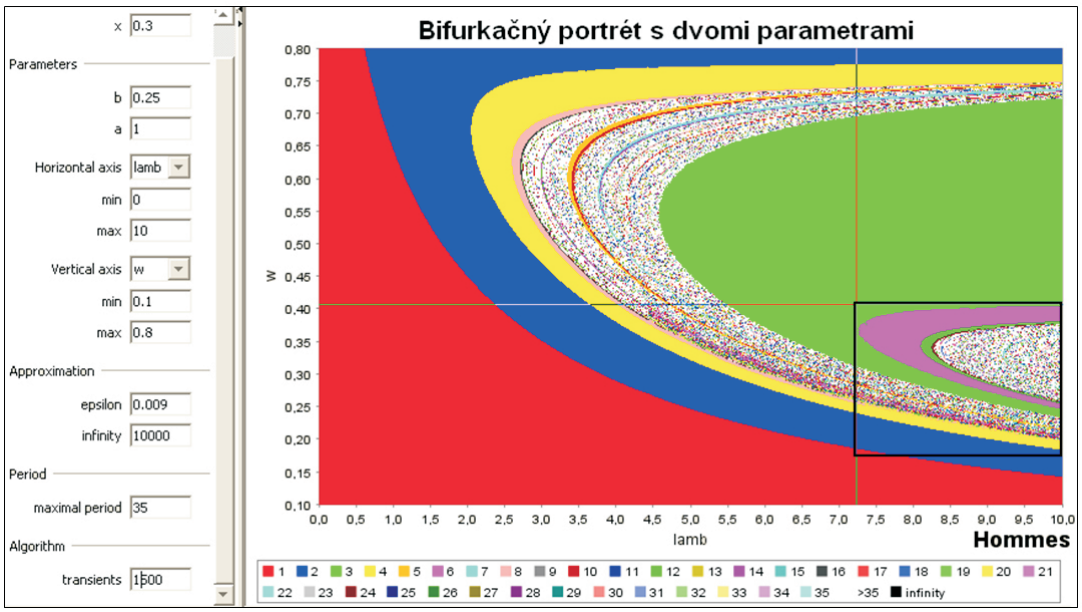
Obr. 22: *Využitie rutiny iDMC Shifted plot- Cobweb animation*



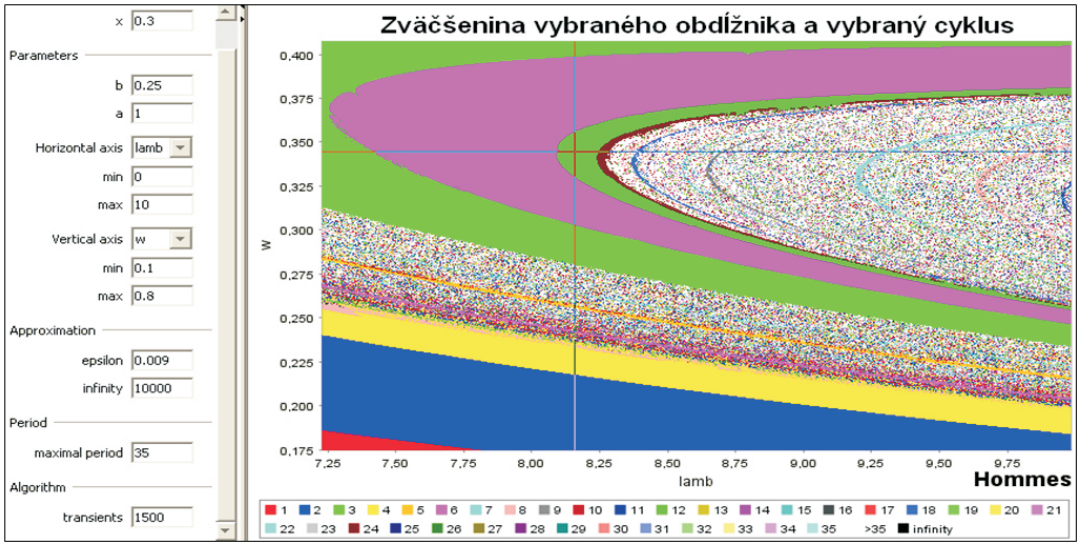
Obr. 23: *Bifurkačný portrét – variovanie s váhovým parametrom w*



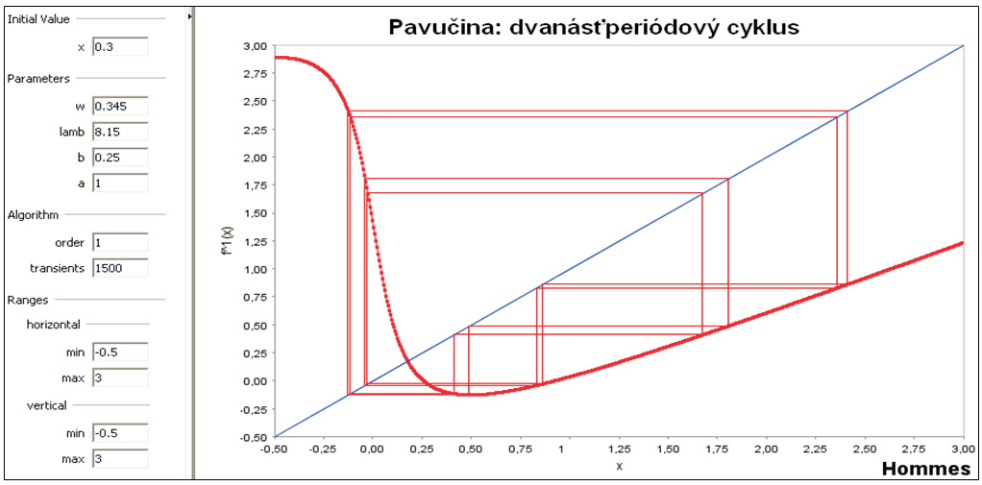
Obr. 24: *Bifurkačný portrét 2D – parametre l a w*



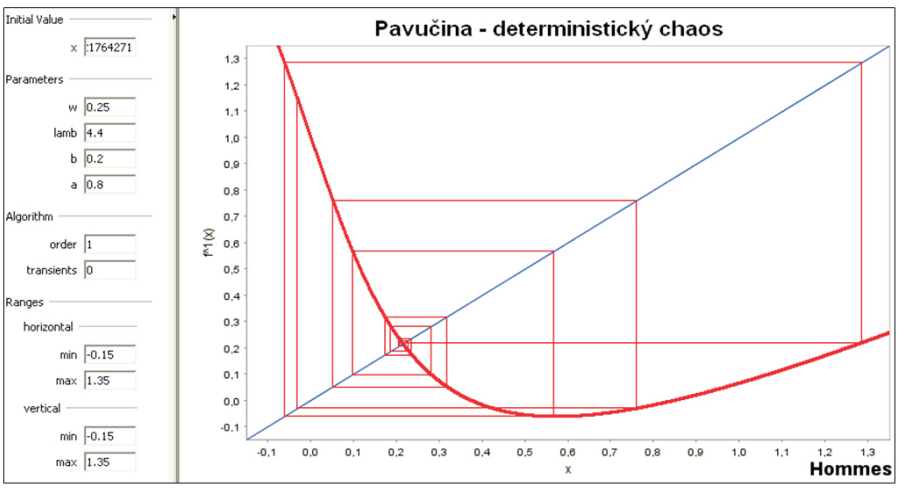
Obr. 25: *Zväčšenina portrétu z obr. 24 a vyznačenie bodu pre experiment*



Obr. 26: *Pavučinový experiment s vybraným bodom z obr. 25*



Obr. 27: *Pavučina s nepárnymi periódami*



V obsahu tohto odseku sme sa zamerali na demonštráciu azda najjednoduchšieho a najčastejšie sa vyskytujúceho mentálneho modelu v elementárnej vysokoškolskej ekonómii. Išlo nám o tom aby sme aj na takomto jednoduchom modeli ilustrovali prednosti IT/CI postupov pri imaginácii komplexných dynamických procesov oproti konvenčným prístupom, resp. sme zároveň predstavili možnosti hybridizácie konvenčnej edukačnej siete VirtLabmi vytvorenými či už učiteľmi alebo, čo je určite ešte lepšie samotnými študentmi v troch prostrediach a to v Excele, v iDMC a v STELLE. Použitie mnohých výstupov z experimentov bolo dôležité preto, aby sa dali dostatočne ilustratívne demonštrovať nevýhody konvenčných metód edukácie oproti prednostiam edukačnej siete a služieb vyzbrojených už aj veľmi jednoduchými produktmi a postupmi IT/CI, nedostatky a nevýhoda tlače na papier vynikne pri sledovaní procesu v konkrétnom virtuálnom laboratóriu.

3. Komparácie modelov a VirtLabov z pohľadu možností hybridizácie siete

Na snímkach obr. 15 až obr. 19 sme na získanie všeobecnej predstavy o entitách vhodných na hybridizáciu edukačných sietí demonštrovali výsledky experimentov na účely komparácie

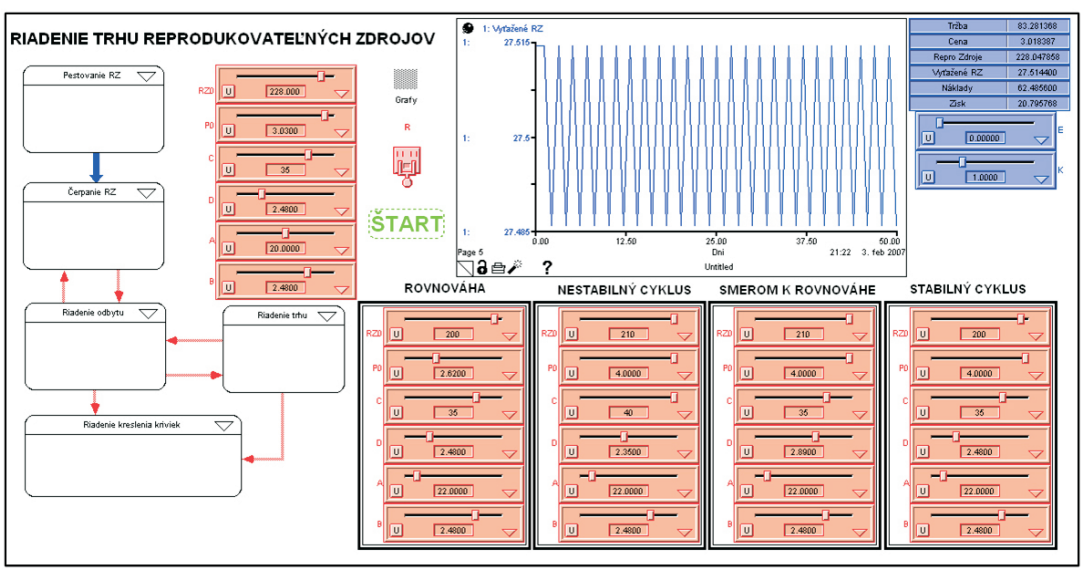
s podobnými experimentmi získanými v Exceli. Rovnako na porovnávanie takých entít sme na snímkach obr. 20 až obr. 27 demonštrovali možnosti niektorých y rutín vstavaných do prostredia iDMC. V tomto odseku predvedieme niektoré podrobnejšie pohľady na prípady, ktoré sme získali aplikáciou postupov a nástrojov STELLA, ako aj iDMC.

Obr. 28: *Licenčný certifikát registrácie STELLY verzie 8.1.1, ktorú sme používali viac rokov*



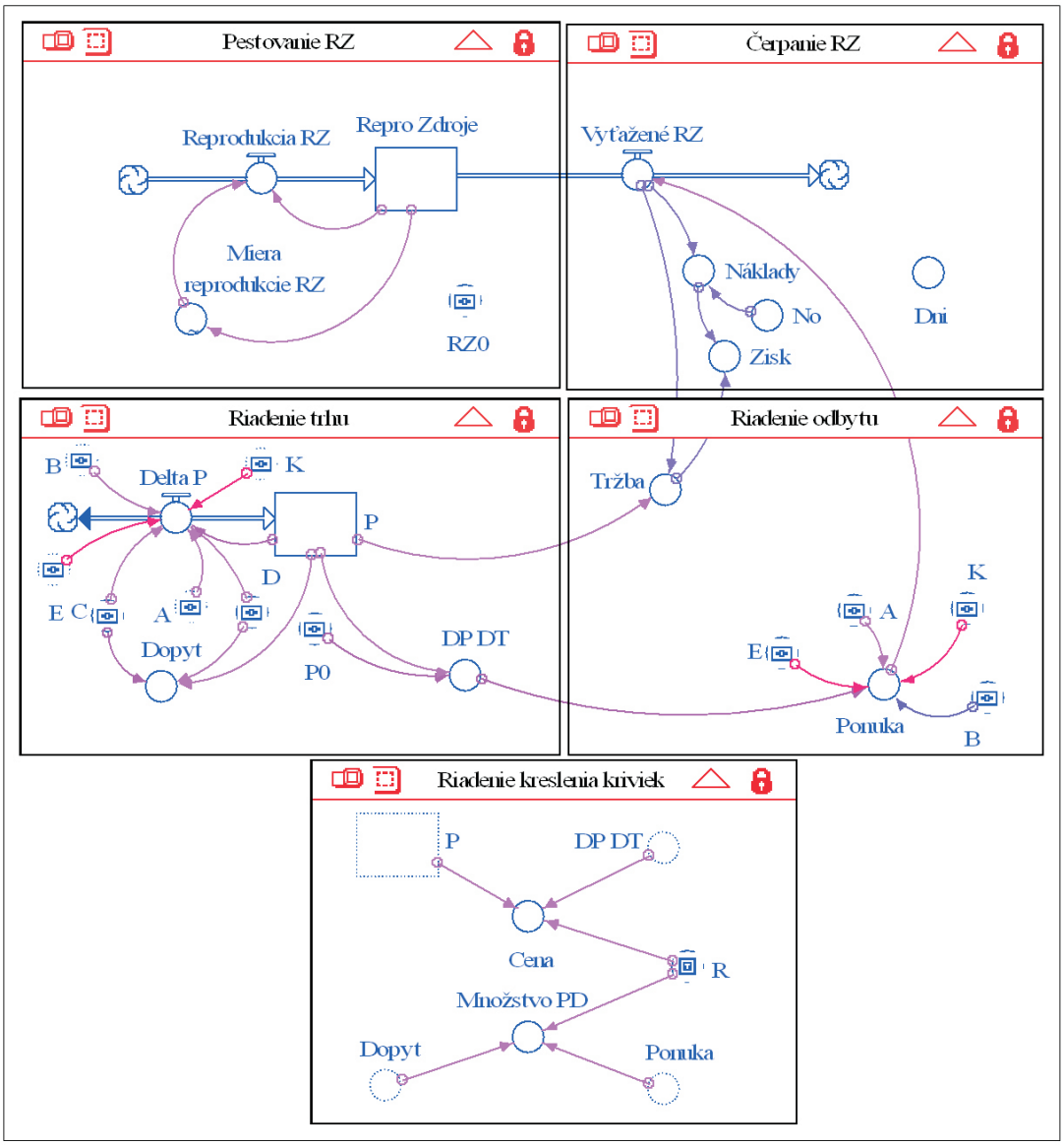
Už skôr sme naznačili, že softvér STELLA ponúka bohatú škálu konštrukčných možností podobného typu ako Simulink MATLAB-u, ale s oveľa jednoduchším a pohodlnejším kreslením blokových schém. Obsahuje mnoho vstavaných funkcií, ako aj rozličné techniky manipulovania grafmi, obrázkami a filmami. Veľkou prednosťou je tiež možnosť programovania príhod, takzvaný

Obr. 29: *Pohľad na hlavnú pracovnú stránku experimentátora, ktorý nie je konštruktérom*



Story-telling realizovaný v podobe živého procesu s priamou možnosťou spúšťania simulačných experimentov. Vzhľadom na obmedzenosť priestoru článku nemožno vo väčšom rozsahu demonštrovať všetky možnosti uvedených softvérov, takže na nasledovných stránkach sú len vybrané ukážky z niektorých produktov autora použitých v edukačnom procese na STU a z prednášok pre doktorandov na EUBA. Na snímkach obr. 28 a 29 je ukážka VirtLabu, ktorý sme vytvorili v STELLE na účely použitia v Story-tellingu o hospodárskom raste a evolúcie na trhu s reprodukovateľnými zdrojmi.

Obr. 29: *Pohľad na stránku konštruktéra modelov*



V nasledovnej trochu širšej demonštrácii opäť aj s prvkami matematiky ukazujeme, že vysokú citlivosť mentálnych modelov cyklického rastu na zmenu parametrov a začiatku (súradnice bodu) evolúcie možno názorne takisto demonštrovať experimentmi vo virtuálnych laboratóriách ak ich predtým než ich vložíme do vybraného softvéru prepíšeme do vhodného matematického formalizmu. Jeden z modelov, ktorý sa veľmi dobre hodí na tento účel sformuloval F. Westerhoff[5] [28], ktorý mierne modifikoval ranú koncepciu P. A. Samuelsona [24]. Ako je

5) Považujeme za korektné poznamenať, že v tejto aktivite nebol sám. Existuje široký zoznam mien, z ktorých však tu uvedieme len niekoľkých z tých, ktorý majú priamy súvis s našou problematikou, takže spomenieme aspoň A. Agliari, L. Gardini, C. Hommes, A. Medio, R. Day, R. Day a P. Chen, T. Puu, J. B. Rosser a I .Sushko.

známe aj z učebníc svoj zárodočný model ekonomického cyklu sformuloval do matematickej podoby P. A. Samuelson ešte v roku 1939 a ako taký sa dá na tento účel použiť aj priamo. Ani on však nebol v tých rokoch sám a patrí sa spomenúť aspoň J. Hicksa, R. Harroda a N. Kaldora. Jeho prístup je diskrétny a pretože berie do úvahy hodnoty v troch časových inštanciách model sa ľahko stane nelineárnym. Symbolika bola v tých časoch už v makroekonómii ustálená, takže národný dôchodok v čase *t*, t. j. Y_t možno písať ako súčet troch zložiek a to spotreby C_t, úspor S_t, či namiesto nich priamo písať indukované súkromné investície I_t a napokon štátne výdavky G_t, takže formálne sa dôchodok rozdelí podľa vzorca

$$Y_t = C_t + I_t + G_t \tag{5}$$

V súlade s Keynesovou analýzou multiplikátora sa predpokladá, že spotrebitelia konzumujú konštantnú časť ich minulého príjmu

$$C_t = aY_t, \text{ resp. pohodlnejšie } C' = aY \tag{6}$$

kde *a* z intervalu $0 < a < 1$ je spriemerovaný marginálny sklon k spotrebe. Nebudeme sa pohoršovať nad nereálnymi predpokladmi či už Keynesa alebo Samuelsona lebo veď nebyť ich, nemali by sme teraz podklady na kritické imaginácie. Ďalší nerealistický predpoklad sa týka investícií. Predpokladá sa, že indukované súkromné investície sú proporcionálne zmenám v spotrebe a teda aj zmenám národného dôchodku. S ľahkosťou im vlastnou sa títo ekonómi povznášajú nad komplexitami, ktoré sa vyskytujú pri bolestnej premene úspor na investície a tie na prírastok kapitálu. Veď už len to, že spotrebiteľ je psychologicky, či etologicky úplne iný subjekt ako investor je dôležitým zdrojom komplikácií a vedie k vážnym fluktuáciám v ekonomickej evolúcii. My však skúmame ich mentálne modely, takže nad nedôslednosťami oných ekonómov sa tiež musíme povzniesť lebo veď potom by naozaj nebolo čo analyzovať. Takže vzorec investícií sa píše v podobe

$$I_t = b(C_t - C_{t-1}) = ab(Y_{t-1} - Y_{t-2}) \tag{7}$$

Z učebníc vieme, že súčin *ab* z intervalu $0 < ab$ označuje kapitálový koeficient, t. j. pomer medzi *zásobou kapitálu* a *tokom dôchodku* je *zafixovaný*. Ďalší heroistický predpoklad, sic., podobne ako fixácia štátnych výdavkov na konštantu, nehovoriac o tom, že sa nerozlišuje medzi neproduktívnymi a produktívnymi výdavkami

$$G_t = \overline{G} \tag{8}$$

Teraz už možno vzorec národného dôchodku prepísať na

$$Y_t = \overline{G} + a(1 + b)Y_{t-1} - abY_{t-2} \tag{9}$$

Tu máme do činenia s rekurentným vzťahom v ktorom hodnoty v postupnosti určuje jednak diferenčná rovnica ale považujeme za dôležité zdôrazniť, že aj spojitá množina potenciálnych začiatočných bodov trajektórií, čo však oní ekonómi ani len netušili. Pre samu diferenčnú rovnicu našli rovnosť, ktorá určuje jedinečný fixný bod (v spojitej analýze sa taký bod nazýva rovnováhou)

$$\overline{Y} = \frac{1}{1-a}\overline{G} \tag{10}$$

Dodajme, že čisto matematicky sa dajú odvodiť aj ďalšie vlastnosti napríklad podmienky stability a cyklický pohyb, ktoré označuje nerovnosť

$$0 < a < \frac{1}{b} < 1 \tag{11}$$

a tiež

$$a < \frac{4b}{(1+b)^2} \tag{12}$$

Pri platnosti nerovností (11) a (12) sa stáva fixný bod *atraktorom* a *stavový bod* ho po istom počte krokov dosiahne *tlmenou osciláciou* spravidla v dvoch periódach aj pri mierne odlišných hodnotách ale väčšie odchýlky vedú k *zdvojovaniu period* až k *deterministickému chaosu*, čo neskôr priamo predvedieme (teda žiaľ iba grafické snímky z výsledkov experimentov, keďže tlačou na papier sme v tom veľmi limitovaní). Ukážeme, že vzájomná súhra a protihra *mediovaná multiplikátorom a akcelerátorom* rodí účinky, ktoré sa prejavujú v *cyklickom raste*, resp. v *komplexnej evolúcii*. V závislosti od zmien parametrov *a*, a *b* môže v ďalšom pokračovaní rekurzu buď rásť alebo sa preklopiť na pokles, či ukážeme vo výsledkoch simulácie pomocou jednotlivých rutín vstavaných do softvéru iDMC, ale až na príklade Westerhoffovho mentálneho modelu.

Modifikácia ktorú zaviedol Westerhoff spočíva v tom, že pridal do modelu očakávania investorov. Potom v jeho koncepcii indukované investície určuje vzťah

$$I_t = b\left(E\left[Y_t\right] - Y_{t-1}\right) \tag{13}$$

To znamená, že indukované investície závisia od rozdielu medzi očakávaným národným dôchodkom v čase *t* a realizovaným národným dôchodkom v čase *t–1*, pričom $b \neq 0$. Otázkou teraz je čo a ako sa vyjadruje formovanie očakávaného dôchodku $E[Y_t]$? *Westerhoff zvolil vyjadrenie via formačného pravidla vážený priemer extrapolačných a regresných očakávaní*

$$E\left[Y_t\right] = W_t E^e\left[Y_t\right] + \left(1 - W_t\right) E^r\left[Y_t\right] \tag{14}$$

a to na základe empirickej evidencie ako to sám zdôvodňuje.

Podľa predstavy Westerhoffa (a nielen jeho) extrapolatívny prediktor je spriemerovaný agent, ktorý očakáva buď *konjunktúru* alebo *recesiu*, čo teda formálne možno zapísať aj takto

$$E^e\left[Y_t\right] = Y_{t-1} + c\left(Y_{t-1} - \overline{Y}\right) \tag{15}$$

kde *c* predstavuje kladný parameter extrapolácie a spoločenský produkt dlhodobej rovnováhy sa považuje za zlomok $\overline{Y} = \frac{\overline{G}}{(1-a)}$, pričom ho však možno považovať aj za voliteľný parameter experimentovania, čo aj neskôr urobíme. V súlade s touto predstavou ak je spoločenský produkt nad (alebo pod) hodnotou $\overline{Y}$, agenti sa správajú optimisticky (alebo pesimisticky) očakávajú, že produkt zostane na vyššej (alebo nižšej) úrovni. Naopak *regresné* očakávanie vyjadruje rovnica

$$E^r\left[Y_t\right] = Y_{t-1} + d\left(\overline{Y} - Y_{t-1}\right) \tag{16}$$

Priemerný *parameter reverze* z intervalu $0 < d < 1$ vyjadruje priemerným agentom očakávanú *adjustačnú rýchlosť* približovania sa hodnoty spoločenského produktu k úrovni dlhodobej rovnováhy $\overline{Y}$. Medzi tieto dva druhy očakávaní Westerhoff zavádza v čase sa meniaci váhový parameter *W*. Chápať ho treba tak, že priemerný agent prikladá tým menší vplyv na extrapolačné pravidlo čím viac sa ekonomika vzďaľuje od dlhodobej rovnováhy $\overline{Y}$. Zdá sa byť prirodzené predpokladať o agentoch, že nevnímajú situácie extrémnych ekonomických podmienok ako trvalé. Relatívny účinok extrapolačného pravidla sa dá vyjadriť rovnicou

$$W_t = \frac{1}{1 + \left(Y_{t-1} - \overline{Y}\right)^2} \tag{17}$$

potom by však relatívny účinok regresných pravidiel mal byť $(1-W_t)$. Kombináciou a patričnou úpravou skôr uvedených vzorcov potom Westerhoff sformuloval rekurenčný vzťah pre premennú národného dôchodku v podobe

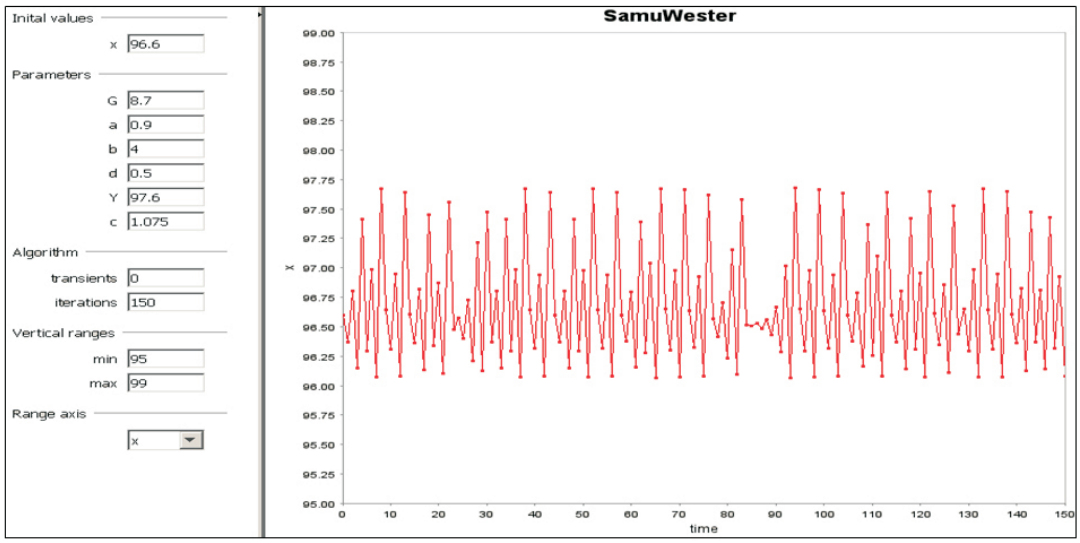
$$Y_t = \overline{G} + aY_{t-1} + \frac{bd\left(\overline{Y} - Y_{t-1}\right) - cd\left(\overline{Y} - Y_{t-1}\right)}{1 + \left(\overline{Y} + Y_{t-1}\right)} \qquad (18)$$

ktorý je 1D nelineárnou diferenčnou rovnicou a za istých podmienok môže byť aj 2D topologickou projekciou inšpirovanou modelom cyklického rastu N. Kaldora, čo budeme demonštrovať neskôr. Vzorec (18) sme tentoraz zase vložili do softvéru iDMC a vytvorili sme tak virtuálne laboratórium pre experimentovanie s modelom Westerhoffa, viď nasledovný výpis

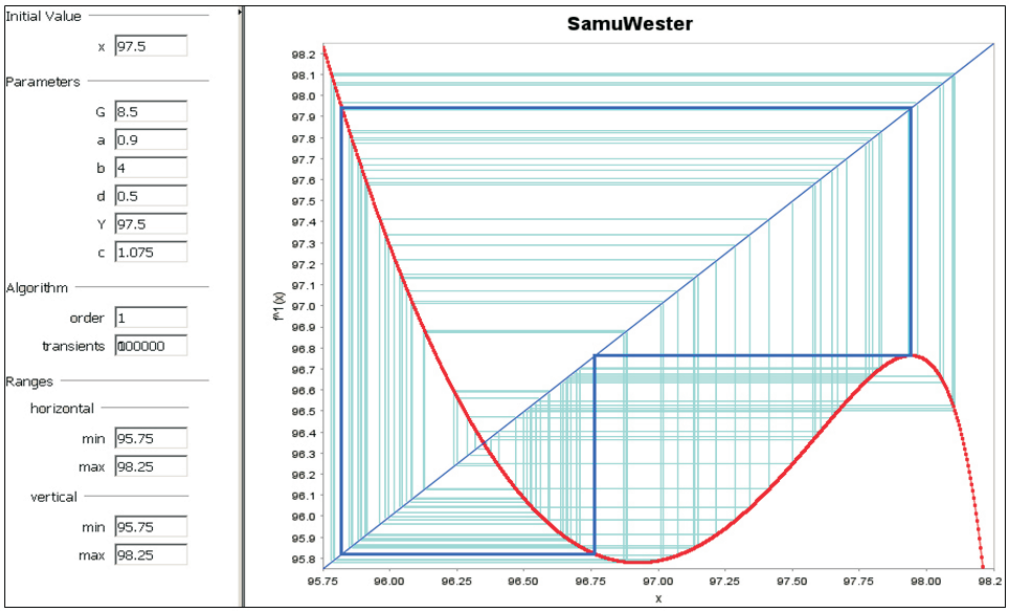
```
—@@
name = "Westerhoff-Samuelson"
description = "Pozri text Andrášik ... )"
type = "D"
parameters = {"G", "a", "b", "d", "Y", "c"}
variables = {"x"}
function f(G, a, b, d, Y, c, x)
   x1 = G + a*x + ((b*d*(Y - x)^3) - c*d*(Y - x))/(1 + (Y - x))
   return x1
end
```

V tomto VirtLabe sme uskutočnili niekoľko experimentov na zviditeľnenie výrazného sklonu mentálneho modelu ku komplexifikácii evolúcie vo virtuálnej realite. Použili sme na to osvedčené rutiny vstavané do iDMC, podobne ako sme to urobili aj v skôr uvedených experimentoch, takže to už nebudem komentovať.

Obr. 30: *Deterministický chaos v podobe časovo-krokovej trajektórie*

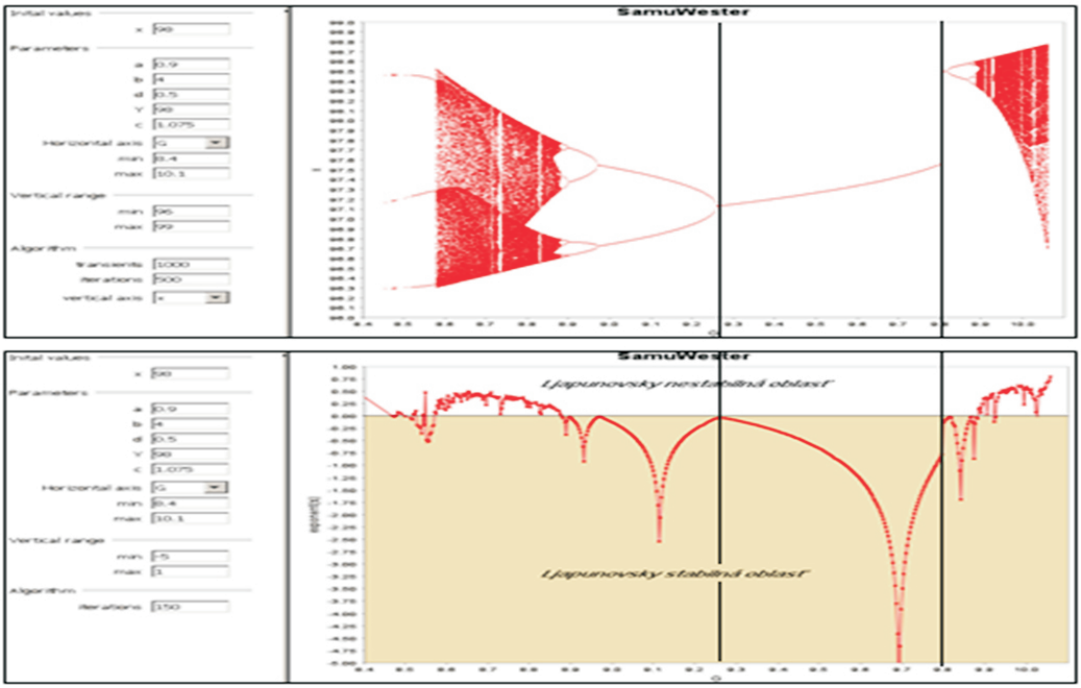


Obr. 31: *Sprvu výrazne deterministicky chaotická pavučina sa uzavrela v 3-periódovom cykle*



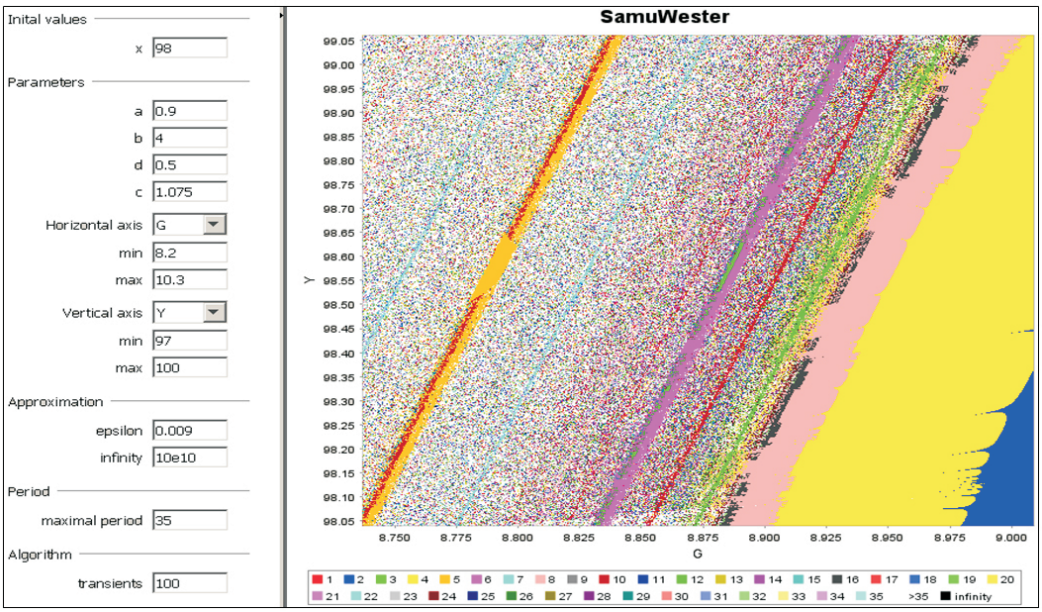
Okrem jednoparametrovej bifurkačnej simulácie sme urobili pokus aj s dvomi parametrami. Bifurkačný portrét označený symbolicky ako $2D_{parameter}$ je priestor v ktorom je kvázi spojitá množina bodov vyjadrená súradnicami dvoch zvolených parametrov. Na snímke obr. 32 je výsledok simulácie získaný rutinou *Bifurcation* v prostredí iDMC.

Obr. 32: *Porovnanie periód zvoleného parametra v bifurkačnom a v Ljapunovovom portréte*

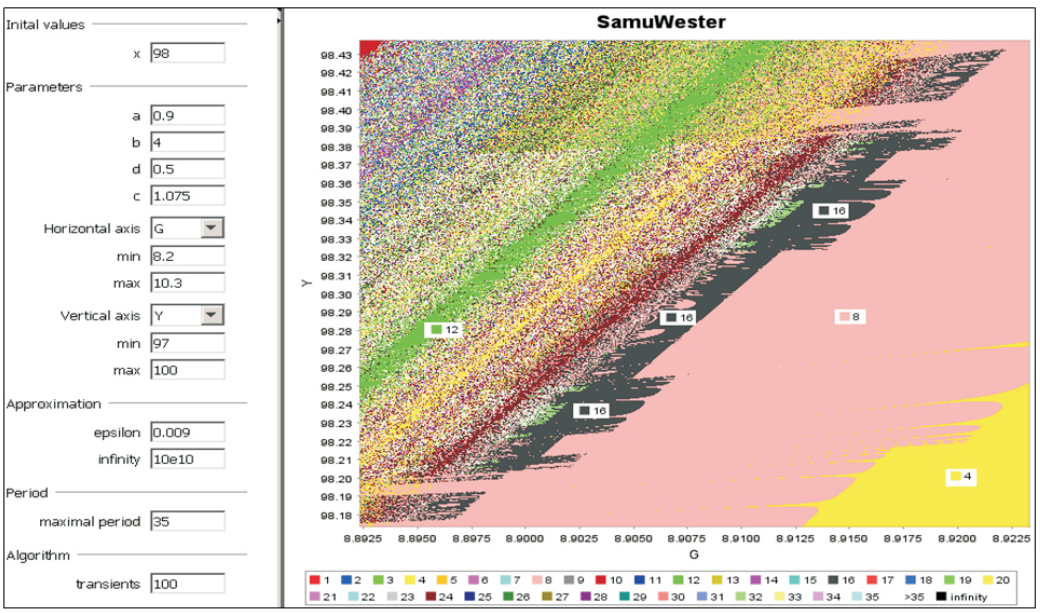


Na snímkach obr. 33-34 predstavujeme možnosti tvorby bifurkačných portrétov s dvoma parametrami. Veľkou výhodou je, že program iDMC umožňuje „mikroskopovanie“ aj pomocou voľby malých obdĺžnikov parameter *1 krát parameter 2*. Takto sa dajú s veľkou presnosťou zistiť bifurkačné oblasti, resp. tzv. Arnoľdove jazyky, ktoré ukazujú kvalitatívne bifurkačné vlastnosti analyzovaného modelu, t. j. periodicitu vyskytujúcu sa v rozličných pomeroch medzi dvoma zvolenými parametrami, ako aj početnosť orbít, resp. pomôžu pri zisťovaní tzv. rotačného čísla pri danom pomere medzi hodnotami parametrov.

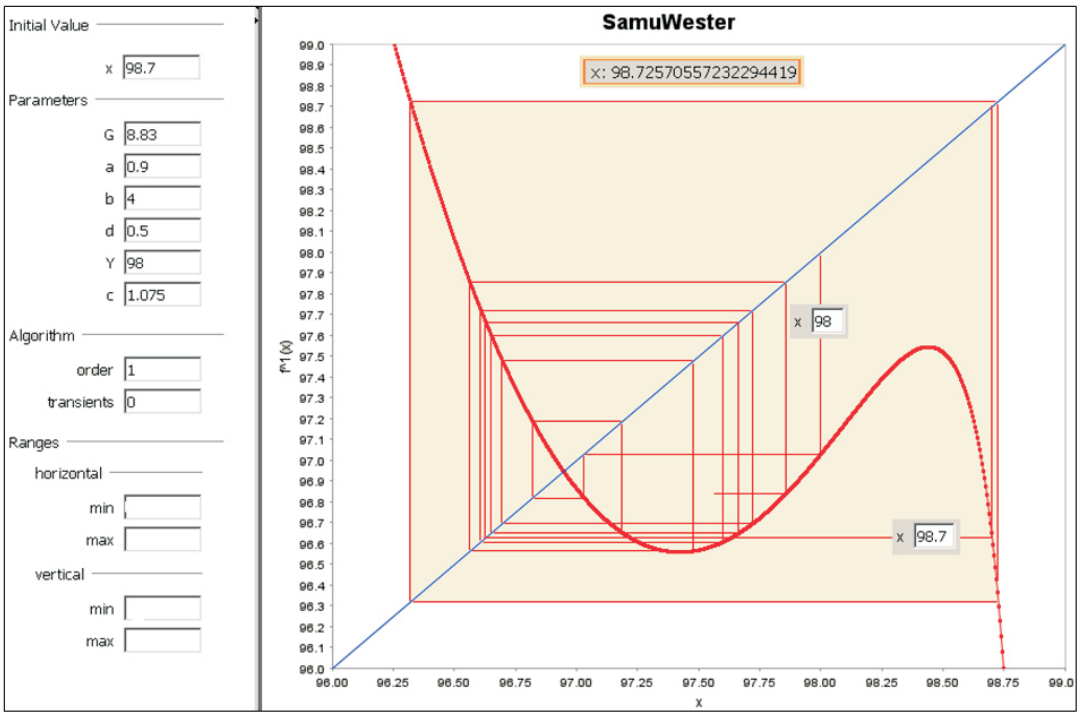
Obr. 33: *Snímka bifurkačného portrétu* $2D_{parameter}$ *G×Y*



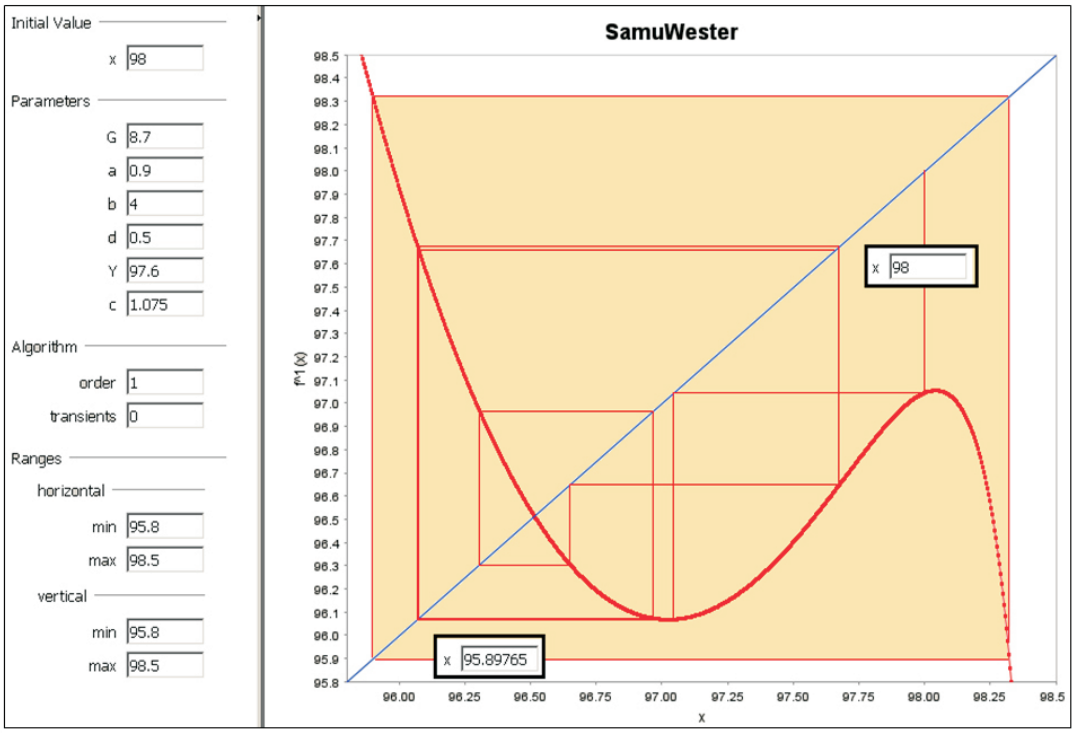
Obr. 34: *Zväčšený malý výsek z bifurkačného portrétu na obr. 4*



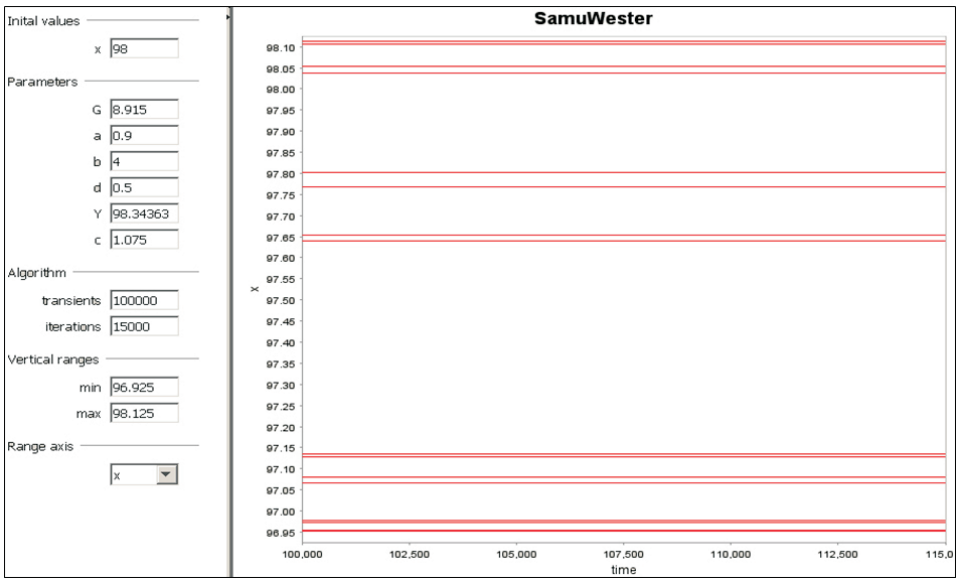
Obr. 35: *Pavučinová mapa s vyfarbenou absorpčnou oblasťou*



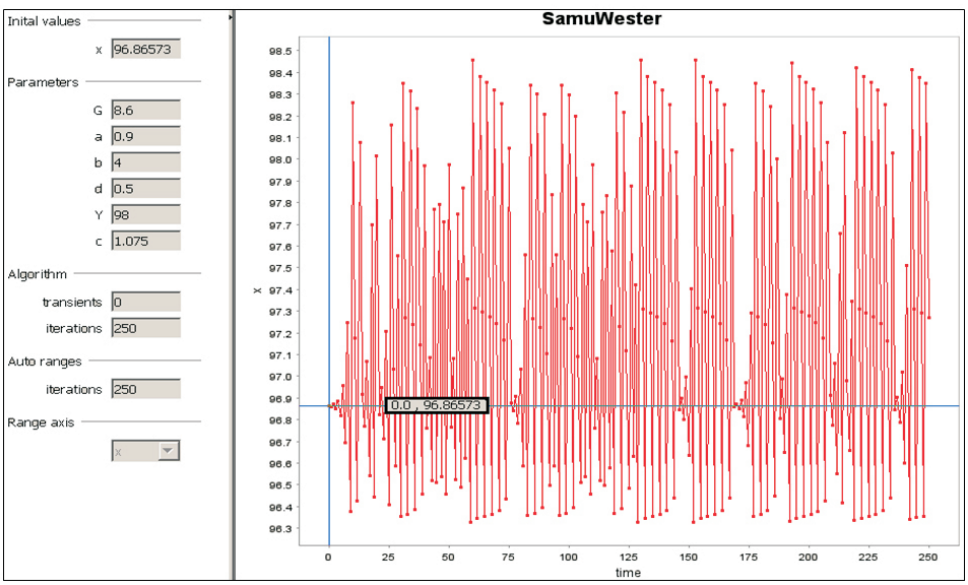
Obr. 36: *Pavučinová mapa so zmenenými hodnotami parametrov*



Obr. 37: *Časovo-kroková trajektória: začiatok vykresľovania po stotisíc krokoch 28 períód*



Obr. 38: *Deterministický chaos: iregulárna rezonancia na úrovni fixného bodu*



Snímky z experimentovania v modeli W-S, ktorý sme realizovali v prostredí iDMC názorne ilustrujú možnosti zvýšenia efektivity a kvality edukačných procesov nielen v ekonómii ale aj v ďalších predmetoch.

Záver

Novšie dosahované pokroky v IT, pokročilé výsledky v komputačnej inteligencii (CI) aj v skupine kognitívnych vied umožňuje zásadným spôsobom zdokonaľovať edukačné siete v ekonomických vedných odboroch. Medzi novými postupmi a metódami majú osobitnú úlohu virtuálne laboratóriá, ktorými sa môžu hybridizovať pôvodne homogénne edukačné siete.

Okrem priamych didaktických prínosov hybridných ekonomických sietí je veľkou výhodou, že súčasné softvéry vhodné na tvorbu virtuálnych laboratórií pre ekonomické študijné programy sú ľahko dostupné, veľmi lacné alebo sú dostupné aj bezodplatne. Ich výhodou je aj to, že k práci s nimi postačia minimálne zručnosti v narábaní s výpočtovou technikou a komunikácii v sieťach.

V prvých semestroch štúdia mávajú študenti ťažkosti pri pochopení komplexných dynamických javov, ktoré sa v súčasnom globálnom vedomostnom svete často vyskytujú a konvenčné metódy im v tom nemôžu veľmi pomôcť. V danej situácii prichádzajú študentom na pomoc IT/CI, Internet a spomínané jednoduché softvéry, ako sú Simulink v MATLABE, SWARM, STELLA, iDMC, ako aj ďalšie im podobné softvéry, samozrejme aj tabuľkový procesor Excel, ktorý je medzi ekonómami obľúbený. Študenti majú jednak možnosť experimentovať s mentálnymi modelmi v dodaných virtuálnych laboratóriách, ale najmä môžu konštruovať vlastné zariadenia pre experimentovanie, čo sa napríklad dá ľahko naučiť v Simulinku a v STELLE. Takto sa študent takmer neintencionálne naučí používať popri inštruktívnych metódach poznávania aj metódy konštruktívne. Autor článku osobitne zameriava pozornosť na to, že všetky tieto nové vymoženosti informačného a vedomostného veku sa ešte znásobia v kvalite a efektívnosti, keď sa zapoja či implementujú do edukačných sietí. Tento hybridizačný proces edukačnej siete ekonómie autor demonštruje niekoľkými jednoduchými príkladmi.

V článku sme sa dotkli aj toho, že využívanie nelineárnych dynamických modelov v kvalitatívnej rovine myslenia v ekonómii ale aj v ďalších ekonomických disciplínach, ako napríklad vo financiách, manažmente, v plánovacích disciplínach atď. sa vo svete v ostatných rokoch rýchlo rozvíja. Žiaľ tento proces však zatiaľ na Slovensku medzi ekonómami neveľmi rezonuje a ešte menej sa používa. Aj odborná a vedecká spisba je v danej oblasti iba sporadická, hoci v ostatných niekoľkých rokoch sa predsa len zaznamenal mierny pokrok ale viac v oblasti rutinného opakovania určitých pragmatických postupov prevzatých „krátkou cestou“ z vyspelejšieho zahraničia. Chceli sme tým naznačiť, že na Slovensku zaostávame najmä v tvorbe vlastných autentických riešení s využitím produktov a služieb IT/CI. Pritom však numerické simulácie kvalitatívnych a fenomenologických ekonomických problémov prinášajú významný pokrok v chápaní takých ekonomických javov a procesov, s ktorými si veda klasickými postupmi nevie dosť dobre poradiť. Dnes už máme pomerne lacno k dispozícii viacero veľmi efektívnych a ľahko ovládateľných softvérových systémov, pomocou ktorých môžeme vytvárať virtuálne ekonomické laboratóriá pre experimentovanie s *komplexnými ekonomickými systémami*.

Z našich mnohoročných skúsenosti s využívaním virtuálnych experimentov v ekonómii a v ďalších ekonomických disciplínach vyplýva poučenie, že treba lepšie a vo väčšej miere používať to čo je v ponuke softvérov bežne dostupné, lacné a ľahko ovládateľné. Príliš sofistikované softvéry komplikujú svojou nevyhnutnou prípravnou fázou študentom ich úsilie a mnohých priamo odrádzajú od používania pokrokových metód získavania poznatkov a vedomostí. Keďže sa mnohé poučky v ekonómii dajú pomerne ľahko formálne naučiť a skúšanie formou vypĺňania testov len zriedka vie odhaliť či študent naozaj rozumie do potrebnej hĺbky danú problematiku využívanie metód ktoré ponúka aplikovaná informatika, komputačná inteligencia a kognitívne vedy zostáva mimo pozornosti nielen študentov ale neraz aj ich pedagógov. Toto tiež prispieva k tomu, že edukácia zostáva na inštruktívnej úrovni odovzdávania „hotových“ poznatkov a tak študenti neinklinujú k tvorivej metóde poznávania vlastným konštruktívnym prístupom k úlohe. Patrí sa na úplný záver korektne dodať, že ani IT/CI obohacovanie konvenčných edukačných sietí v ekonomických študijných programoch nemožno považovať za panaceum na všetky neduhy, a ťažkosti v procese ekonomického vzdelávania, ale naše mnohoročbné skúsenosti potvrdzujú ich úspešnosť v kvalite aj efektivite porozumenia komplexných dynamických procesov.

Použitá literatúra[6]:

- [1] Allen, P. M. (1988). Dynamic models of evolving systems, System Dynamics Review, 4, pp. 109-130.
- [2] Andrášik, L. (2004).Teória počítačového experimentovania v umelom hospodárstve, Ekonomický časopis, 52, č. 8, s. 996.
- [3] Andrášik, L. (2006). Experience with Digital Story-telling in Social Sciences Education, Proceedings of the 7th International Symposium on Computational Intelligence, Hungarian Fuzzy Association, Budapest.
- [4] Andrášik, L. (2008). Digitálne príbehy v nelineárnych dynamických ekonomikách v diskrétnom čase, Ekonomický časopis, 56, č. 3, s. 239.
- [5] Andrasik, L. (2008). Virtual Laboratories as a Medium for Knowledge Acquisition in a Complex Problems, Proceedings of Virtual university – International conferrence, Bratislava, in internet: http://virtuni.eas.sk/ 2008/pdf/fid001351.pdf.
- [6] Andrasik, L. (2010). Computational Qualitative Economics: Computational Intelligence--Assisted Building, Writing and Running Virtual Economic Theories, in: Computational Intelligence in Engineering, I. J. Rudas, J. Fodor, J. Kacprzyk (Eds.), p. 246-261, Springer-Verlag, Berlin, Heidelberg.
- [7] Arthur,W. B. (1993). On Designing Economic Agents that Behave Like Human Agents, Journal of Evolutionary Economics 3, pp. 1-22.
- [8] Arthur, W. B. (1997). Durlauf, S. N. and Lane, D. A., Eds., The Economy as an Evolving Complex System, II, Proceedings Volume XXVII, Santa Fe Institute Studies in the Sciences of Complexity, Addison-Wesley, Reading, MA,
- [9] Arthur, W. B., Positive Feedbacks in the Economy, Scientific American, p. 92-99, 1990.
- [10] Coase, R. H. (1940). The Analysis of Producers' Expectations. *Economica*, 7, 280-292.
- [11] Ezekiel, M. (1938). The Cobweb Theorem. *Quarterly Journal of Economics*, 52, 255-280.
- [12] Hannon, B., Ruth M. (1997). Modeling Dynamic Economic Systems, Springer-Verlag, New York,
- [13] Luna, F., et all. (1997). Economic Simulation in SWARM: Agent-based Modelling and Object Oriented Programming, Kluwer Academic Publishers, Boston, 2000.
- [14] Hicks, J. R.: A Contribution to the Theory of the Trade Cycle, Oxford University Press, Oxford, 1950.
- [15] Hommes, C. H. (1994). Dynamics of the cobweb model with expectations and nonlinear supply and adaptive demand, Journal of Economic Behaviour&organisation, 24, pp. 315-335.
- [16] Hommes, C., H. (1995). A reconsideration of Hicks' nonlinear trade cycle model, Structural Change and Economic Dynamics, 6, p. 435–59.
- [17] Lines, M. and Westerhoff, F. (2006). Expectations and the multiplieraccelerator model, forthcoming, in Puu T. and Sushko I., Springer.
- [18] Medio, A. and Lines, M. Nonlinear Dynamics: a Primer, Cambridge University Press, Cambridge. (2001).
- [19] Mira, C., Gardini, L., Barugola, A., Cathala, J. C.: Chaotic Dynamics in Two-Dimensional Noninvertible Maps, series A, vol. 20, World Scientific Publishing, Singapore, 1996.
- [20] Puu, T., Gardini, L. and Sushko, I.: A Hicksian multiplier–accelerator model with floor determined by capital stock, Journal of Economic Behaviour and Organization, forthcoming, 56, p. 331–348, 2004.
- [21] Puu, T.: The Hicksian trade cycle with floor and ceiling dependent on capital stock, Journal of Economic Dynamics & Control, v. 31, p. 575–592, 2007.

6) Do zoznamu literatúry sme zaradili kvôli korektnosti aj autorov, ktorých v článku priamo necitujeme, ale o výsledky ich prác sa v našej vedeckej aj pedagogickej činnosti dlhodobo opierame.

- [22] Puu, T. (1997). Nonlinear Economic Dynamics, Springer-Verlag, Berlin.
- [23] Rosser, J. B.: From Catastrophe to Chaos – A General Theory of Economic Discontinuities, 2nd edn, Kluwer Academic Publishers, Boston 2000.
- [24] Samuelson, P. A.: Interactions between the multiplier analysis and the principle of acceleration, Review of Economic Statistics, 21, p. 75–8, 1939.
- [25] Thom, R. (1972). Structural Stability and Morphogenesis, W. A. Benjam.
- [26] Vannini, A. (2005). From mechanical to life causation, Syntropy, Vol. 1, p. 80-105.
- [27] Zeeman, E. C. (1974). On the unstable behaviour of stock exchanges. Journal of Mathematical Economics 1, s.39–49.
- [28] Westerhoff, F. H.: Samuelson's multiplier–accelerator model revisited, Applied Economics Letters, 13, p. 89–92, 2006.

prof. Ing. Ladislav Andrášik, DrSc.
Ekonomická univerzita v Bratislave
Dolnozemská 1/b, 852 35 Bratislava
ladislav.andrasik@euba.sk

Modelový aspekt analýzy ekonomických procesov podniku a metódy riešenia

Model aspect of the analysis of companies economic processes and methods for their solutions

Ivan Brezina, Zuzana Čičková

Abstract:

In the analysis of economic processes is generally appropriate and effective to apply a quantitative approach based on the application of mathematical models. Economic-mathematical models provide the ability to model the economic process through mathematical expressions and relations. The aim is to formulate such models and related problems that can be addressed by appropriate mathematical methods. The corresponding problems can be solved using standard optimization methods. Because of the problems arising from the needs of practice often leads to large-scale models and large computational complexity, to apply such methods is sometimes impossible, or difficult, time consuming, so now the increasing interest is turned to non-traditional methods, especially in the field of artificial intelligence. Effective fulfillment of the nature of the model approach and solution of the resulting optimization problems using a classical, resp. "modern" methods is possible only in conjunction with the use of appropriate tools of information and communication technologies.

Keywords:

modelling, economic-mathematical model, optimization, computational complexity, heuristics

ACM classification: F.2, F.4.1, G.0, I.2.8, J.4

ÚVOD

Ekonomické procesy predstavujú vo všeobecnosti veľmi komplikované systémy a preto podrobné vyjadrenie ich vnútorných relácií, ako aj relácií na okolie, možno realizovať len prostredníctvom pomerne zložitého systému vzťahov, ktoré možno vyjadriť pomocou zložitého systému modelov, resp. prostredníctvom zložitých parciálnych modelov. Na analýzu ekonomických procesov možno pritom veľmi efektívne použiť kvantitatívne prístupy založené na aplikácii matematických modelov.

Matematická analýza zameraná na kvantitatívne vzťahy medzi jednotlivými ekonomickými a niekedy aj neekonomickými faktormi, ktoré sa dynamicky vyvíjajú v čase a vzájomne na seba interaktívne pôsobia, poskytuje možnosť aspoň čiastočne odhaliť ich vnútorné závislosti. Pri skúmaní jednotlivých ekonomických procesov nie je zvyčajne potrebné detailne analyzovať informácie o jeho všetkých aspektoch. Relevantné je predovšetkým vymedzenie hlavných

aspektov a vzťahov daného procesu. Namiesto reálneho ekonomického procesu možno potom skúmať jeho zjednodušené vyjadrenie prostredníctvom modelu.

Za najvšeobecnejšie, ale aj najabstraktnejšie, možno považovať ekonomicko-matematické modely, ktoré prezentujú modelovaný ekonomický proces prostredníctvom matematických výrazov a vzťahov. Modelované veličiny možno vyjadriť pomocou sústavy premenných, parametrov a konštánt. Cieľom modelového prístupu je formulovať také modely a s nimi súvisiace úlohy, ktoré možno riešiť vhodnými matematickými metódami a prostriedkami. Na ich riešenie sa využívajú klasické optimalizačné metódy riešenia. Pretože úlohy vyplývajúce z potrieb praxe často vedú k modelom veľkého rozsahu a veľkej výpočtovej zložitosti, riešenie takýmito metódami býva niekedy nemožné, respektíve komplikované a zdĺhavé, v súčasnosti preto možno sledovať zvyšujúci sa záujem o netradičné metódy riešenia, predovšetkým z oblasti umelej inteligencie. Efektívne naplnenie podstaty modelového prístupu a riešenie z toho vyplývajúcich optimalizačných úloh pomocou klasických, resp. „moderných" metód je možné len v súčinnosti s využitím vhodných prostriedkov informačných a komunikačných technológií.

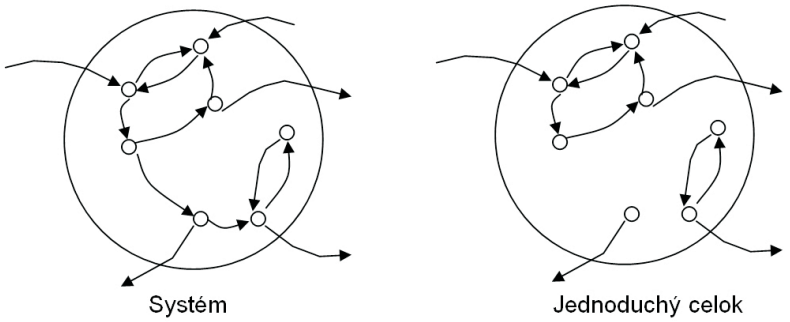
1. Problém, model, úloha

Súčasná transformácia ekonomických procesov a nové smery vo vývoji informačných a komunikačných technológií prinášajú odklon od tradičných prístupov k riadeniu a zmenu nárokov na podporné informácie. Distribúcia riadenia a funkcií do viacerých uzlov vedie k posilneniu decentralizovaných organizačných štruktúr, ktoré sú schopné aplikovať rýchle zmeny v celom systéme. Presunom rozhodovacích právomocí do miest vzniku informačnej potreby možno podstatne skrátiť čas rozhodovania, minimalizovať potrebu komunikácie a dosiahnuť určitú autonómiu pri rozhodovaní tak, že na rozhodovaní sa podieľajú len tie prvky systému, ktorých sa problém týka (Závodný, Chudá, 2006).

K tomu, aby mala organizácia k dispozícii relevantné individuálne informácie, musí mať k dispozícii vhodné informačné systémy. Nevyhnutné je pritom získavať podstatné a overené dáta, ktoré sú nevyhnutné pre rozhodovanie. Rozloženie dát na viaceré uzly v systémoch distribuovaného spracovania dát vedie k možnosti prevádzkovať jednotlivé podsystémy informačného systému organizácie, čo spôsobuje minimalizáciu toku dát, zvýšenie výkonu a zlepšenie spoľahlivosti systému (Goga, 2005).

Z hľadiska systémového chápania ekonomických javov, napr. (Sakál, Jerz, 2006), (Šimkovic, Terek, 1987), (Goga, 2005), je vhodné chápať ekonomický systém ako celok vzájomne prepojených prvkov, ktoré na seba navzájom priamo alebo nepriamo pôsobia, pričom ekonomický systém vždy zameraný na dosiahnutie nejakého špecifického ekonomického cieľa, prípadne skupiny cieľov. Všeobecne má systém tieto základné vlastnosti:

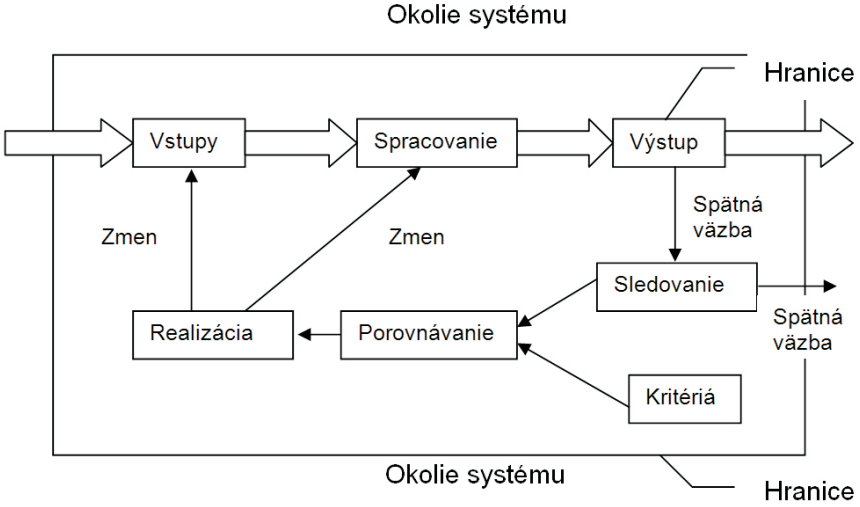
1. celostnosť a čiastočnosť – systém predstavuje celostné zoskupenie prvkov, systém teda strany možno chápať ako celok, ale na druhej strane možno v tomto celku identifikovať samostatné prvky, ktoré môžu existovať len v rámci systému, je to teda jednotný celok vzájomne sa ovplyvňujúcich prvkov, ktoré sú často nehomogénne, ale spolupatričné (obr. 1),
2. zviazanosť – medzi prvkami systému možno uplatniť rozličné vzájomné väzby, ktoré sú práve zo systémového hľadiska podstatné (tým sa líši systém od súhrnu prvkov),
3. organizovanosť – systém možno charakterizovať jeho vnútornou organizáciou,
4. integračné vlastnosti – systému ako celok možno identifikovať na základe vlastnosti, ktorými nedisponuje osobitne ani jeden jeho prvok, pretože prvok systému je len jeho časťou a nie celkom, integračné vlastnosti spôsobujú, že systém ako celok nadobúda synergický efekt – dosahuje vyššiu efektívnosť ako je súčtová efektívnosť jeho častí (jeden z prvých teoretikov teórie systémov Ludwig von Bertanlaffy skúmal, prečo sa úspešné systémy prejavujú tak, že celok je významnejší ako súčet všetkých prvkov dohromady akoby platilo 2 + 2 = 5).



Obr.1: *Systém a jednoduchý celok*

Systém sa vo všeobecnosti vždy skladá z jednotlivých systémových elementov, špecifických pre každý osobitný systém. Zvyčajne sa ako hlavné systémové elementy uvádzajú, napr. (Šimkovic, Terek, 1987), (Goga, 2005): vstupy, kritériá, spracovanie, výstupy, spätná väzba, sledovanie, porovnávanie, realizácia (obr. 2).

Vstupy zvyčajne predstavujú hmotné a nehmotné toky, ktoré vstupujú z okolia do systému. Ďalej sa zameriame na nehmotné vstupy do systému, ktoré predstavujú informácie. Kritériá sú definované cieľmi, ktoré má systém dosiahnuť. Stanovením kritérií je determinovaný výkon systému. Vyhodnotenie kritérií umožňuje posúdiť, či systém pracuje dobre, či úspešne plní všetky svoje funkcie. Toto vyhodnotenie má podstatný vplyv na riadenie celého systému a môže sa premietnuť napríklad do úpravy vstupných tokov. Tieto informácie, ktorý majú spätný vplyv na systém, možno charakterizovať ako spätnú väzbu. Spracovanie predstavuje vlastnú premenu vstupov na výstupy, ktorá závisí od charakteristiky systému, ďalej budeme sledovať proces premeny primárnych údajov na informácie. Výstupy sú finálnym produktom systému. Sledovanie vo svojej podstate znamená meranie výstupov systému (množstvo, kvalitu a pod.) a teda sledovanie plnenia kritérií (cieľov) systému. Porovnávanie predstavuje prostriedok na vyhodnocovanie dosiahnutých výsledkov v súvislosti s naplánovanými hodnotami. Na základe porovnávania možno zisťovať odchýlky od naplánovaného stavu a v rámci realizácie tieto odchýlky upravovať.



Obr. 2: *Hlavné systémové elementy*

Systém je vo všeobecnosti abstraktný pojem, predstavuje spôsob myslenia o objekte, resp. procese, ktorý reálne existuje (Goga, 2005). Systémový prístup poskytuje široké možnosti na modelovanie ekonomických procesov, ktorý vychádza na jednej strane z koncepcie skúmania ekonomických javov, na druhej strane akceptuje rozvoj matematických prístupov z rôznych

aplikačných oblastí Sojka, Walter a kol., 1986. Vo všeobecnosti sa pod klasickým modelovaním chápe formulácia modelu, numerické aspekty riešenia s ním súvisiacej úlohy, problém zabezpečenia údajov a otázka spoľahlivosti údajov, otázka nasadenia a využitia výpočtovej techniky a implementácia modelu ((Sojka, Walter a kol., 1986)). Rozšírené chápanie modelového prístupu pre ekonomické systémy poskytujú (Kiš, Čileg, Vugdelija, Sedlak, 2005), ktorý pod tvorbou ekonomicko-matematického modelu chápu zber dát (vytvorenie bázy údajov), formuláciu ekonomicko-matematického modelu, riešenie modelu, ekonomickú interpretáciu, kvalitatívnu analýzu riešenia a formulovanie návrhov pre hospodársku rozhodovanie, rozhodnutie, realizáciu rozhodnutia a kontrolu realizácie, analýzu ďalších návrhov a informáciu o potrebe preformulovania modelu. Iný prístup na podporu rozhodovacích procesov predkladajú (Chajdiak, Grell, 2006), ktorí rozlišujú pojmy model a úloha a proces modelovania zhŕňajú do identifikácie a definície problému (definovanie cieľa), riešenia problému rozhodovania – tvorby modelu (konštrukcia rozhodovacej úlohy), výber riešenia a interpretácia získaných výsledkov s rozhodnutím, realizácia prijatého rozhodnutia. Podobný prístup je uplatnený v prácach (Ivaničová, Brezina, Pekár, 2002), (Brezina, Ivaničová, Pekár, 2007). Zhrnutím rôznych prístupov by mohla byť nasledujúca konštrukcia:

1. Identifikácia problému a definovanie cieľa, výber faktorov a obmedzujúcich podmienok.
2. Analýza objektu, identifikácia jeho vlastností.
3. Tvorba modelu, vytvorenie funkčného modelu (analýza požiadaviek, analýza informácií, stanovenie kritéria, stanovenie variantov) a dátového modelu (zisťovanie údajov, odhad rizika).
4. Konštrukcia rozhodovacej úlohy.
5. Hľadanie metód a výber metódy na riešenie rozhodovacej úlohy.
6. Nájdenie, resp. tvorba zodpovedajúceho algoritmu na riešenie rozhodovacej úlohy.
7. Výber programu na riešenie rozhodovacej úlohy a jej riešenie.
8. Verifikácia samotného modelu.
9. Simulácie realizované na rozhodovacej úlohe (simulácie predovšetkým extrémnych prípadov).
10. Interpretácia získaných výsledkov a prijatie rozhodnutia.
11. Realizácia prijatého rozhodnutia.

V súlade s uvedenou ideovou schémou treba definovať základné pojmy. Pod pojmom *problém* možno všeobecne rozumieť vymedzenie obsahu rozhodovacieho procesu obsahujúceho podstatné parametre. *Model* predstavuje viac alebo menej presný zjednodušený odraz reality (integruje v sebe najdôležitejšie zložky reálneho systému). *Matematický model* predstavuje zjednodušené formalizované zobrazenie reálnych vzťahov pomocou matematických nástrojov, teda identifikáciu štruktúry tvarov matematických vzťahov a identifikáciu parametrov reprezentovaných konkrétnymi číselnými hodnotami (ekonomicko-matematický model zobrazuje relevantné vzťahy ekonomického procesu). Formalizácia problému matematickými prostriedkami spočíva vo vytvorení určitej sústavy matematických vzťahov s konkrétnymi premennými, spravidla sa vyžaduje adaptácia štandardných postupov, ale aj individuálny prístup analytika schopného identifikovať zodpovedajúci, resp. vytvoriť nový typ modelu. Na formalizáciu možno aplikovať aj iný, ako matematický model (napr. grafický model), na počítačové riešenie je však matematická formulácia najvhodnejšia. *Rozhodovacia úloha* je potom konkrétne zadanie všetkých parametrov modelu (model predstavuje napr. lineárny model optimalizácie výrobného procesu podniku, rozhodovaciu úlohu konkrétny výrobný proces v určitom čase, s určitými vstupnými údajmi). Na riešenie rozhodovacej úlohy treba predovšetkým identifikovať a následne použiť vhodný zodpovedajúci *algoritmus*, ktorý predstavuje presný výpočtový postup, ktorý pre každé zadanie parametrov nájde riešenie danej rozhodovacej úlohy. Pod algoritmom možno teda rozumieť postupnosť krokov, resp. pravidiel, ktoré pre každé zadanie parametrov nájdu riešenie danej rozhodovacej úlohy alebo poskytnú informáciu, že príslušné riešenie neexistuje.

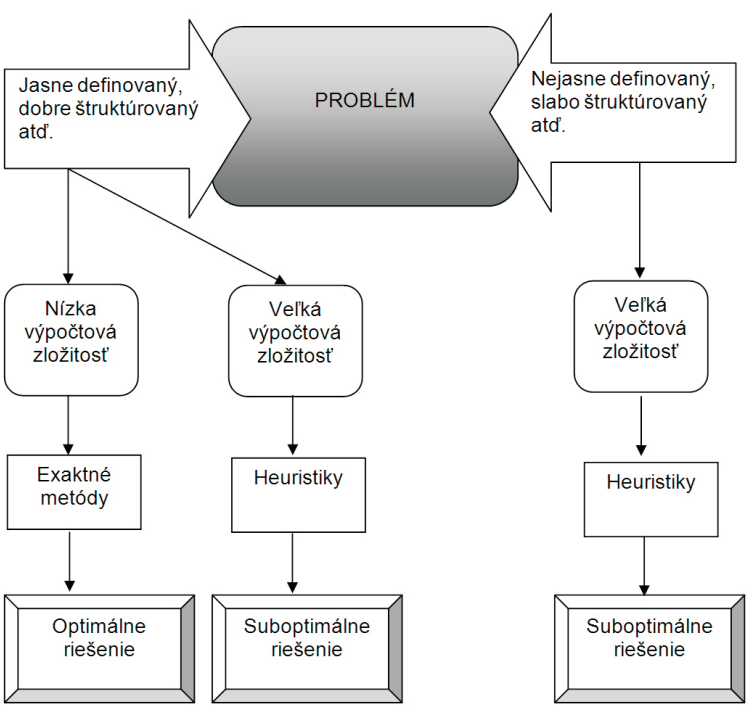
2. Metódy riešenia rozhodovacích úloh

Optimalizácia ekonomických problémov prostredníctvom riešenia rozhodovacích úloh skonštruovaných v rámci ekonomicko-matematických modelov všeobecne vyžaduje identifikáciu extrému funkcie na množine, ktorá je ohraničená rôznymi ohraničujúcimi podmienkami, pričom definičný obor premenných funkcie môže nadobúdať rôzny charakter (celočíselný, reálny, komplexný, diskrétny, atď.), a premenné optimalizovanej funkcie musia spĺňať rôzne obmedzujúce podmienky (Čičková, 2007).

Riešenie optimalizačných rozhodovacích úloh závisí predovšetkým od miery štruktúrovanosti optimalizovaného problému, od veľkosti a výpočtovej zložitosti riešenej rozhodovacej úlohy. Pre dostatočne štruktúrovaný model a na ňom definovanú rozhodovaciu úlohu s nízkou výpočtovou zložitosťou je výhodnejšie na jej riešenie použiť exaktné metódy, ktoré zaručujú dosiahnutie optimálneho riešenia. Ak napr. z časového hľadiska nemožno využiť exaktné metódy, na riešenie rozhodovacích úloh možno použiť heuristické metódy, resp. metódy umelej inteligencie.

Klasifikácia metód na riešenie optimalizačných rozhodovacích úloh sa zvyčajne v literatúre, napr. (Unčovský, 1985) uvádza v štruktúre:

- Exaktné – výsledkom je nájdenie optimálneho riešenia rozhodovacích úloh, ak existuje, avšak tieto metódy sú zvyčajne použiteľné len pre dostatočne štruktúrované rozhodovacie úlohy s nízkou výpočtovou zložitosťou,
- Heuristické – nezaručujú získanie optimálneho, niekedy ani len prípustného, riešenia rozhodovacej úlohy, poskytujú však často dobré riešenie v prípadoch, keď výpočet exaktného riešenia nie je možný, predovšetkým pre nedostatočne štruktúrované rozhodovacie úlohy s vysokou výpočtovou zložitosťou,
- Kombinované – pôvodná rozhodovacia úloha sa zvyčajne dekomponuje na rôzne podúlohy, z ktorých jedna množina je riešená exaktnými metódami a druhá množina použitím heuristík.



Obr. 3: *Možnosti použitia exaktných techník a metód umelej inteligencie*

Rozdiel medzi exaktnými metódami a heuristickými metódami (heuristikami) spočíva predovšetkým v tom, že heuristiky nemusia zabezpečiť nájdenie optimálneho a niekedy ani prípustného riešenia rozhodovacej úlohy (Čičková, 2007). Výhoda heuristických metód však spočíva v skutočnosti, že za pomerne krátky čas môžu poskytnúť dobré riešenie aj pre nedostatočne štruktúrované rozhodovacie úlohy s vysokou výpočtovou zložitosťou, ktoré sa zvyčajne označuje pojmom suboptimálne riešenie. Heuristika je zvyčajne konštruovaná tak, aby využívala špecifické vlastnosti konkrétnej rozhodovacej úlohy. Heuristiky sa pre svoju jednoduchosť a rýchlosť často využívajú na nájdenie východiskového riešenia, na ktoré je potom v ďalšej etape výpočtu aplikovaný exaktný algoritmus s cieľom nájdenia optimálneho riešenia.

Pre algoritmy umelej inteligencie na riešenie optimalizačných rozhodovacích úloh možno použiť klasifikáciu heuristík:

- klasické heuristiky (deterministické algoritmy založené len na štandardných matematických metódach),
- moderné heuristiky, resp. metaheuristiky (algoritmy na báze umelej inteligencie).

Medzi najdôležitejšie metódy umelej inteligencie z aplikačného hľadiska patria (Janíček – Ondráček, 1998):

1. Metódy, pre ktoré sú v ich aplikačnom procese charakteristické tieto fázy (napr. znalostné systémy a tzv. neurónové siete):
   - fáza učenia sa – tzv. „prázdnemu systému" tvorenému len matematickými štruktúrami je predkladaná množina známych skutočností súvisiacich s riešeným problémom,
   - fáza dotazov – tzv. „naučenému" systému sú kladené otázky z oblasti, v ktorej bol „učený",
   - fáza odpovedí – systém je schopný odpovedať na otázky na základe naučených faktov.
2. Metódy, ktoré pri riešení problému využívajú prehľadávanie stavového priestoru riešení s cieľom nájsť globálny extrém určitej optimalizovanej funkcie, ktorej matematické vyjadrenie musí byť známe. K týmto metódam patria napr. genetické algoritmy a simulované chladenie.
3. Hybridné metódy – prepojenie a doplnenie jednotlivých metód umelej inteligencie.

Dôvodom, prečo je v súčasnosti popri existujúcich analytických prístupoch a numerických metódach venovaná veľká pozornosť aj prístupom umelej inteligencie, je fakt, že metódy na báze umelej inteligencie sú použiteľné aj na riešenie rozhodovacích úloh, pre ktoré nie je známy efektívny algoritmus na ich riešenie, ale tiež aj v prípade, keď správanie sa systému možno vyjadriť napríklad len slovne. Pretože pre mnoho matematických problémov nie je doposiaľ známa ich matematická riešiteľnosť, resp. známe existujúce algoritmy nemajú dobré matematické vlastnosti (sú nestabilné numericky alebo vzhľadom na vstupné údaje), použitie algoritmov umelej inteligencie predstavuje jednu z možností riešenia rozhodovacích úloh s veľkou výpočtovou zložitosťou, resp. rozhodovacích úloh postrádajúcich „žiaduce" matematické vlastnosti.

3. Výpočtová zložitosť

Teória výpočtovej zložitosti (napr. Brezina, Gežík, Čičková, 2012) je v súčasnosti rozvinutou vednou disciplínou s pomerne zložitým obsahom, ktorá umožňuje rozlíšiť medzi úlohami „ľahko" a „ťažko" riešiteľnými a rozhodnúť o vhodnom algoritme použiteľnom na riešenie zodpovedajúcej optimalizačnej rozhodovacej úlohy, pretože od správneho výberu závisí kvalita získaného riešenia. Vlastnosti niektorých dobre štrukturovaných rozhodovacích úloh umožňujú ich optimálne riešenie aj pre pomerne veľké rozmery (príkladom sú úlohy lineárneho programovania), riešenie niektorých rozhodovacích úloh je už omnoho zložitejšie a identifikácia optimálneho riešenia pre rozhodovacie úlohy veľkých rozmerov nie je možná ani s využitím súčasnej

najvýkonnejšej výpočtovej techniky (napr. väčšina kombinatorických úloh s binárnymi premennými).

Veľký význam preto nadobúda analýza vlastností rozhodovacej úlohy, na základe možno zvoliť efektívny algoritmus na jej riešenie. Zvoleným algoritmom je determinovaná kvalita získaného riešenia, pričom je nutné brať do úvahy existujúce výpočtové možnosti. Zvolená metóda riešenia musí zabezpečiť dosiahnutie požadovaných výstupných informácií (výsledkov) a presne určiť, ktoré vstupné údaje budú potrebné. Výber vhodného algoritmu je zvyčajne determinovaný časom, ktorý je k dispozícii na formuláciu a riešenie rozhodovacej úlohy.

Po formulácii a analýze rozhodovacej úlohy treba opísať logiku a presný postup riešenia rozhodovacej úlohy (logický postup inštrukcií zabezpečujúcich riešenie). Výsledkom je tzv. riešiaci algoritmus, ktorého praktickou implementáciou je zvyčajne zodpovedajúci počítačový program.

Na riešenie akejkoľvek konkrétnej rozhodovacej úlohy možno vo všeobecnosti použiť niekoľko algoritmov. Efektívnosť ich použitia je determinovaná ich časovou a pamäťovou zložitosťou, ktoré závisia od rozmeru úlohy, teda od počtu všetkých parametrov (vstupných dát – n)[1)]. *Časová zložitosť* udáva, ako pri zväčšovaní vstupných dát rastie objem času potrebného na riešenie rozhodovacej úlohy. Možno ju určiť ako čas, ktorý potrebuje algoritmus na riešenie rozhodovacej úlohy v závislosti od jej rozmeru, meria sa nepriamo počtom elementárnych krokov algoritmu. *Pamäťová zložitosť* udáva, ako pri zväčšovaní vstupných dát rastie objem pamäte potrebnej na riešenie úlohy.

Algoritmus považovať za lepší, čím kratší čas a menší objem pamäte potrebuje na nájdenie riešenia danej rozhodovacej úlohy. Zjednodušene možno posudzovať algoritmus podľa počtu elementárnych krokov, ktoré sú považované za rovnako časovo náročné (napr. sčítanie, odčítanie, násobenie, delenie, porovnávanie vetvením atď.) a ktoré treba vykonať na vyriešenie danej rozhodovacej úlohy. Vo všeobecnosti však čas závisí nielen od veľkosti rozhodovacej úlohy, ale aj od konkrétnej implementácie algoritmu. Preto je zvyčajne opísaná rýchlosť algoritmu na základe rýchlosti, s akou rastie čas potrebný na vyriešenie rozhodovacej úlohy vzhľadom na jej rozmery.

Vo všeobecnosti je zaujímavý vzťah času potrebného na preverenie záznamov k veľkosti vstupných dát n. Hľadá sa teda funkcia, ktorá vyjadruje celkový čas potrebný na realizáciu príslušnej postupnosti krokov, teda funkcia f, ktorá pre rozsah vstupných údajov veľkosti n určí hodnotu $f(n)$, t. j. čas výpočtu pre vstup danej veľkosti. Zvyčajne sa odhaduje časovú náročnosť výpočtu len rádovo, teda zisťuje sa, či je funkcia lineárna, kvadratická, logaritmická, exponenciálna a pod. V tabuľke 1 sú uvedené odhady hodnôt niektorých funkcií pre zväčšujúce sa hodnoty n.

Tabuľka 1: *Odhad hodnôt niektorých funkcií*[2)]

N	10	50	100	300	1000
Polynomiálne funkcie					
$\ln(n)$	2,3	3,91	4,61	5,7	6,91
$n\ln(n)$	23,02	195,6	460,52	1711,14	6907,76
$n\log_2 n$	33	282	665	2469	9966
$5n$	50	250	500	1500	5000
n^2	100	2500	10000	90000	10^6
$5n^2$	500	12500	50000	450000	5.10^6
$30n^2$	3000	75000	300000	$2{,}7.10^6$	30.10^6
n^3	1000	125000	10^6	$2{,}7.10^7$	10^9

1) Avšak v súčasnosti je čoraz jasnejšie, že paralelný čas a veľkosť hardvéru sú tiež dôležité miery zložitosti.

2) Na výpočet hodnôt bola použitá internetová služba Wolfram Alpha http://www.wolframalpha.com/

Nepolynomiálne funkcie					
2^n	1024	$1{,}13.10^{15}$	$1{,}27.10^{30}$	$2{,}04.10^{90}$	$1{,}07.10^{301}$
e^n	22026,47	$5{,}18.10^{21}$	$2{,}69.10^{43}$	$1{,}94.10^{130}$	$1{,}97.10^{434}$
$n!$	$3{,}63.10^{6}$	$3{,}04.10^{64}$	$9{,}33.10^{157}$	$3{,}06.10^{615}$	$4{,}02.10^{2567}$
n^n	10^{10}	$8{,}88.10^{84}$	10^{200}	$1{,}37.10^{743}$	10^{3000}

Odhad trvania $f(n)$ operácií, ak je na vykonanie jednej potrebný čas 1 μs (10^{-6}s) uvádza tabuľka 2[3)].

Tabuľka 2: *Odhad trvania f(n) operácií, ak jedna vyžaduje čas 1 μs*

N	10	50	100	300	1000
Polynomiálne funkcie					
$\ln(n)$	$2{,}3.10^{-6}$ s	$3{,}91.10^{-6}$ s	$4{,}61.10^{-6}$ s	$5{,}7.10^{-6}$ s	$6{,}91.10^{-6}$ s
$n\ln(n)$	$2{,}3.10^{-5}$ s	$1{,}96.10^{-4}$ s	$4{,}61.10^{-4}$ s	$1{,}71.10^{-3}$ s	$6{,}91.10^{-3}$ s
$n\log_2 n$	$3{,}3.10^{-5}$ s	$2{,}82.10^{-4}$ s	$6{,}65.10^{-4}$ s	$2{,}47.10^{-3}$ s	$9{,}97.10^{-3}$ s
$5n$	5.10^{-5} s	$2{,}5.10^{-4}$ s	5.10^{-4} s	$1{,}5.10^{-3}$ s	5.10^{-3} s
n^2	10^{-4} s	$2{,}5.10^{-3}$ s	10^{-2} s	9.10^{-2} s	1 s
$5n^2$	5.10^{-4} s	$1{,}25.10^{-2}$ s	5.10^{-2} s	0,45 s	5 s
$30n^2$	3.10^{-3} s	$7{,}5.10^{-2}$ s	0,3 s	2,7 s	30 s
n^3	10^{-3} s	0,125 s	1 s	27 s	16,67 min.
Nepolynomiálne funkcie					
2^n	$1{,}02.10^{-3}$ s	35,83 rokov	$4{,}03.10^{16}$ rokov	$6{,}47.10^{76}$ rokov	$3{,}39.10^{287}$ rokov
e^n	0,02 s	$1{,}64.10^{8}$ rokov	$8{,}53.10^{29}$ rokov	$6{,}15.10^{116}$ rokov	$6{,}25.10^{420}$ rokov
$n!$	3,63 s	$9{,}64.10^{50}$ rokov	$2{,}96.10^{144}$ rokov	$9{,}7.10^{601}$ rokov	$1{,}28.10^{2554}$rokov
n^n	2,78 hod.	$2{,}82.10^{71}$ rokov	$3{,}17.10^{186}$rokov	$4{,}34.10^{729}$rokov	$3{,}17.10^{3729}$rokov

Z tabuľky 2 je zrejmé, že pri polynomiálnych algoritmoch možno dosiahnuť výsledky aj pre relatívne veľké úlohy, ale napr. algoritmus, ktorý vyžaduje čas 2^n, je nepoužiteľný už pre $n = 50$ a algoritmus, pre ktorý je potrebný čas dokonca $n!$, je ešte zložitejší (prakticky nepoužiteľný už pre $n = 19{,}19! \approx 1{,}22.10^{17}$, trvanie v μs: $19!.10^{-6} \approx 1{,}22.10^{11}$ $\mu s \approx 121645$ *s* ≈ 2027 min ≈ 33,8 hod. ≈ ≈ 1,4 dňa)[4)].

Ani zdokonaľovanie výpočtovej techniky neznamená súčasne veľký pokrok riešenia algoritmami s exponenciálnou zložitosťou. Pri riešení rôznych optimalizačných rozhodovacích úloh sa nemožno spoliehať len na vývoj v oblasti výpočtovej techniky, ale súčasné smerovanie sa zameriava skôr na vývoj efektívnejších algoritmov.

Na vyriešenie dvoch úloh rovnakej veľkosti môže algoritmus vyžadovať rôzny čas, pričom je nezanedbateľným faktorom aj to, že čas vykonania rovnakých operácií v rôznych programových prostrediach a na rôznych počítačoch môže mať rôzne trvanie. Preto sa uvažuje o tzv. *asymptotickom raste* funkcie $f(n)$, ktorý sa odhaduje pre veľké hodnoty n bez ohľadu na koeficienty funkcie. Funkcia asymptotického rastu sa nazýva *časová zložitosť*, resp. *trieda* algoritmu a označuje sa symbolom θ. Pri asymptotickej zložitosti sa uvažuje len o raste funkcie $f(n)$ pre úlohy veľkých rozmerov. Ak uvažujeme napr. funkciu $f(n)$ $n^2 + 3n + 4$ a funkciu $f(n)$ $3n^2 + 2$, *pri asymptotickej zložitosti sa uvažuje len s najviac rastúcim členom, teda* n^2, zanedbávajú sa pomalšie rastúce členy a konštanty, pre obidve funkcie je zložitosť $\theta(n^2)$.

3) Odhadovaný počet protónov viditeľného vesmíru je 79-ciferné číslo a počet mikrosekúnd od „veľkého tresku“ je asi 24-ciferné číslo ($1{,}5.10^{25}$ sekúnd).

4) Pri n=20 je odhadovaný čas 24 dní.

Ak za $f(n)$ možno dosadiť polynóm $p(n)$, algoritmus nazývame *polynomiálny algoritmus,* ak nie, ide o exponenciálny algoritmus. Odhad zložitosti týchto tried algoritmov je v tabuľke 1.

Pre polynomiálny algoritmus (rýchly algoritmus) je čas na spracovanie vstupných dát veľkosti n ohraničený zhora funkciou n, $\log n$, n^2, n^3 a pod. Existuje teda taký polynóm $p(n)$, pre ktorý počet krokov, ktoré algoritmus vykoná pri výpočte, je ohraničený číslom $p(n)$. Potom algoritmus má zložitosť $\theta(p(n))$, pre polynóm $p(n)$.

Pre exponenciálny algoritmus (pomalý algoritmus)[5] je čas na spracovanie vstupných dát veľkosti n ohraničený zhora funkciou typu $2n$, $n!$, n^n atď. Odhad počtu krokov, ktoré algoritmus vykoná pri výpočte, je prinajmenšom exponenciálny. Menej zložitý je teda ten algoritmus, pre ktorý je funkcia $f(n)$ funkciou s menším rastom. Napríklad algoritmus, ktorý pracuje v čase $\theta(\log(n))$, je lepší a rýchlejší ako ten, čo pracuje v čase $\theta(n)$ a ten je lepší ako algoritmus, ktorý pracuje v čase $\theta(n^3)$. Tiež pre každý polynóm $p(n)$ sú algoritmy pracujúce v čase $\theta(p(n))$ rýchlejšie ako napríklad algoritmy pracujúce v čase napr. $\theta(2^n)$ alebo dokonca $\theta(n!)$. Dôvodom zavedenia pojmu asymptotickej zložitosti (nárastom funkcií, t. j. „správaním" funkcie od vysokej hodnoty n) je práve porovnanie algoritmov. Nedostatky porovnania podľa asymptotickej zložitosti sú však zrejmé. Napr. algoritmus, ktorý pracuje v čase n^3, je pomalší ako algoritmus pracujúci v čase 2^n pre vstupy veľkosti $n \leq 9$ a algoritmus pracujúci v čase $1000n$ je pomalší ako algoritmus vyžadujúci čas n^2, pre všetky vstupy veľkosti $n \leq 999$ (tabuľka 2). To znamená, že v praxi je najlepším prístupom kombinácia oboch typov, t. j. pre úlohy malých rozmerov sa niekedy javí efektívnejšie použiť algoritmus s vyššou zložitosťou.

V teórii zložitosti algoritmov sa rozlišujú triedy tzv. P-úloh a tzv. NP-úloh. Formálne najpresnejšia definícia P a NP úloh je definícia pomocou nedeterministického Turingovho stroja alebo aspoň nedeterministického algoritmu.

Predmetom teórie zložitosti sú zvyčajne tzv. rozhodovacie úlohy, pričom známejšie sú ich optimalizačné verzie. Cieľom riešenia optimalizačnej verzie úlohy je nájsť také riešenie, ktoré je optimálne z hľadiska určitého kritéria, resp. skupiny kritérií. Cieľom riešenia rozhodovacích verzií úloh je len konečné konštatovanie „áno" alebo „nie". Je zrejmé, že riešenie optimalizačnej verzie úlohy nie je ľahšie ako riešenie jej rozhodovacej verzie. Ak nájdeme najlepšie riešenie, vieme, či je aspoň také dobré ako daná hodnota K. Možno však dokázať, že ak je rozhodovacia verzia riešiteľná polynomiálne, je aj optimalizačná verzia riešiteľná polynomiálne. V dôkaze tohto tvrdenia sa využíva fakt, že skladaním, násobením a sčítaním polynómov dostaneme opäť polynómy[6].

Trieda úloh, ktorých rozhodovacia verzia je algoritmicky riešiteľná v polynomiálnom čase, sa označuje ako trieda P. Algoritmy použiteľné na riešenie takýchto úloh možno označiť za algoritmy „rýchle", t. j. pri úlohách tohto typu netreba použiť heuristický algoritmus, pretože možno rýchlo a efektívne získať optimálne riešenie.

Existuje však aj trieda úloh, pre ktoré nie je dosiaľ známy algoritmus pracujúci v polynomiálnom čase, avšak existuje nedeterministický algoritmus, ktorý pri správnom výbere umožní ich riešenie v polynomiálnom čase (nedeterministický algoritmus sa líši od deterministického tým, že v niektorých krokoch výpočtu sa možno ľubovoľne rozhodnúť pre niektorý z niekoľkých postupov). Trieda úloh, ktoré možno riešiť nedeterministickým polynomiálnym algoritmom, sa označuje ako trieda NP.

Deterministický algoritmus je špeciálnym typom nedeterministického algoritmu a preto platí $P \subset NP$. Otázka, či platí $P = NP$ alebo $P \neq NP$, je jednou z najdôležitejších nezodpovedaných

5) Je potrebné poznamenať, že za exponenciálne sú považované algoritmy s nepolynomiálnym rastom, napr. $(\lg n)^{\lg n}$, aj keď v matematike sa táto funkcia za exponenciálnu nepovažuje.

6) Fakt, že nedeterministický algoritmus pracuje v polynomiálnom čase, znamená, že každý z krokov vyžaduje polynomiálny čas a tiež aj reťazec s musí mať najviac polynomiálnu dĺžku vzhľadom na veľkosť úlohy.

otázok teoretickej informatiky[7]. Predpokladá sa, že tieto triedy sa nerovnajú, pretože vlastnosť polynomiálnej redukcie zaručuje, že by existovali pre stovky dosiaľ opísaných NP úloh polynomiálne algoritmy. To sa však doteraz nepodarilo dokázať.

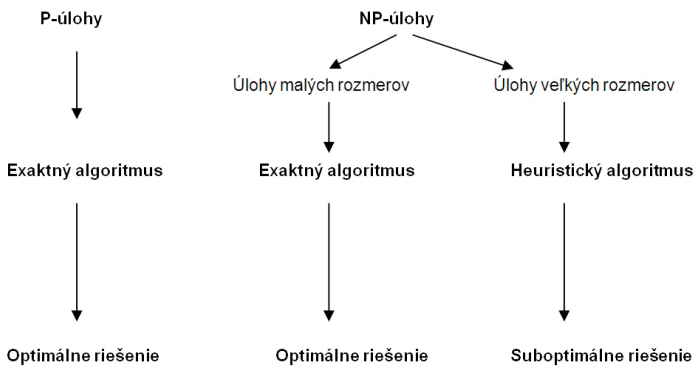
Z triedy NP možno ďalej vymedziť tzv. NP-úplné úlohy. Rozhodovacia úloha je NP-úplná, ak je NP úlohou a každá NP-úloha sa na polynomiálne redukuje. Táto vlastnosť zaručuje, že nájdenie polynomiálneho algoritmu pre jednu úlohu by znamenalo riešenie všetkých ostatných úloh z triedy NP-úplné v polynomiálnom čase.

Ak je pri definícii NP-úplnej úlohy vynechaná požiadavka, aby úloha bola NP-úlohou, získavame tzv. NP-ťažkú úlohu. Úloha je teda NP-ťažká, ak sa na ňu polynomiálne redukuje každá NP-úloha (NP ťažká úloha nemusí byť v triede NP úplná). Ak by sme teda našli algoritmus na riešenie akejkoľvek NP-ťažkej úlohy, vedeli by sme vyriešiť každú NP-úplnú úlohu.

Záver

Problematiku analýzy ekonomických procesov a riešenia zodpovedajúcich rozhodovacích úloh je aj v súčasnosti aktuálnou problematikou. Preto je potrebné rozvíjať systémový prístup k modelovaniu ekonomických procesov, ktorý vychádza na jednej strane z koncepcie skúmania ekonomických javov, na druhej strane akceptuje rozvoj matematických prístupov z rôznych aplikačných oblastí. Dôležitou súčasťou systémového prístupu k analýze ekonomických procesov, ktorý zahŕňa identifikáciu problému, analýzu objektu a identifikáciu jeho vlastností, vytvorenie modelu, konštrukciu rozhodovacej úlohy, nájdenie metódy na jej riešenie, nájdenie, resp. vytvorenie zodpovedajúceho algoritmu, výber programu a riešenie rozhodovacej úlohy, verifikáciu modelu, následné simulácie, interpretáciu získaných výsledkov a prijatie rozhodnutia a jeho následnú realizáciu je využitie alebo vytvorenie zodpovedajúceho algoritmu.

Pri riešení úloh triedy P je pomerne ľahké využiť optimalizačný algoritmus s polynomiálnou zložitosťou, ktorý poskytuje optimálne riešenie pomerne rýchlo a umožňuje aj riešenie rozhodovacích úloh veľkého rozsahu (napr. úloha lineárneho programovania, nájdenie najkratšej cesty v grafe, nájdenie maximálneho toku v grafe atď.). Riešenie úloh triedy NP (úlohy celočíselného, resp. binárneho programovania, problém obchodného cestujúceho atď.) už také jednoduché nie je. Pri riešení NP-úloh možno postupovať dvojakým spôsobom, možno použiť niektorú z exaktných metód, čo pravdepodobne povedie k exponenciálnemu algoritmu (explicitná enumerácia, dynamické programovanie), alebo možno vytvoriť heuristický algoritmus, ktorý bude polynomiálny, ale nemusí vždy viesť k optimálnemu riešeniu. Situácia je znázornená na obrázku 4.



Obr. 4 *Použitie rôznych druhov metód na riešenie P- a NP-úloh*

7) Otázka, či P = NP bola zaradená medzi tzv. sedem problémov milénia, ktorých zoznam bol zostavený na Clay Mathematics Institute, za vyriešenie každého z nich je riešiteľovi garantovaná odmena 1 000 000 $.

Rozhodnutie o použití vhodného algoritmu na riešenie úlohy spočíva predovšetkým v schopnosti analyzovať riešenú rozhodovaciu úlohu, určiť jej príslušnosť k triede P, resp. NP a potom na základe jej rozmeru a možnosti použitia výpočtovej techniky určiť algoritmus na riešenie uvedenej rozhodovacej úlohy v závislosti od času, ktorý je k dispozícii na jej riešenie.

Použitá literatúra:

- [1] ACKOFF, R.L., SASIENI, M.: *Foundations of Operations Research*. New York: John Willey, 1968.
- [2] BREZINA, I., IVANIČOVÁ, Z., PEKÁR, J.: *Operačná analýza.* Bratislava: IURA Edition, 2007.
- [3] BREZINA, I., GEŽÍK, P., ČIČKOVÁ, Z.: *Sieťová analýza.* Bratislava: Ekonóm, 2012.
- [4] COOK, A. S.: *Prehľad teórie výpočtovej zložitosti.* Pokroky matematiky, fyziky a astronomie, Vol. 32 (1987), No. 1, 12—29.
- [5] ČIČKOVÁ, Z.: *Metódy operačného výskumu a umelej inteligencie pri riešení optimalizačných problémov*. In Ekonomika a informatika: Bratislava, 2007. ISSN 1336-3514. – Roč. 5, č. 1 (2007), s. 154-162.
- [6] GOGA, M.: *Kvantitatívne metódy a riadenie*. Bratislava: SÚVAHA, 2005. Vydané pre Slovenskú komoru certifikovaných účtovníkov. ISBN 80-88727-90-1
- [7] HAUPT, L. R., HAUPT, S. E.: *Practical Genetic algorithm*. New Jersey: John Wiley&Sons, Inc. Publication, 2004.
- [8] HILLIS, D.: *Obrazce v kameni*. Bratislava: Kalligram, 2002.
- [9] CHAJDIAK, J., GRELL, M.: *Podpora rozhodovacích procesov*. Bratislava: Statis, 2006.
- [10] IVANIČOVÁ, Z., BREZINA, I.,PEKÁR, J.: *Operačný výskum.* Bratislava: IURA Edition, 2002.
- [11] JANÍČEK, P., ONDRÁČEK, O.: *Řešení problémů modelováním*. Brno: PC-DIR Real, s.r.o., 1998.
- [12] KIŠ, T., ČILEG, M., VUGDELIJA, D., SEDLAK, O.: *Kvantitativni metodi u* ekonomiji. Subotica: Ekonomski fakultet, Novi Sad: Offset print, 2005.
- [13] KOBBACY, K. A. H., VADERA, S., RASMY, M. H.: *AI and OR in management of operations: history and trends*. Journal of the Operational Research Society 58, 2007
- [14] MAŘÍK, V., ŠTĚPÁNKOVÁ, O., LAŽANSKÝ, J: *Umělá intelligence 3*. Praha: Academia, 2001.
- [15] ONWUBOLU, G. C., BABU, B. V: *New Optimization Techniques in Engineering. Studies in Fuzziness and Soft Computing*. Berlin, Heidelberg, New York, Hong Kong, London, Milano, Paris, Tokyo Sringer, Volume 14, 2004.
- [16] PALÚCH, S.: *Algoritmická teória grafov*. Žilina: ŽU v Žiline, 2008. Dostupné na: http://frcatel.fri.uniza.sk/users/paluch/graf.pdf.
- [17] SAKÁL, P., JERZ, V.: *Operačná analýza v praxi manažéra II*. Trnava: SP SYNERGIA, 2006.
- [18] SEKAJ, I: *Evolučné výpočty a ich využitie v praxi*. Bratislava: IRIS, 2005
- [19] SOJKA, J., WALTER, J. a kol.: *Matematické modelovanie ekonomických procseov*. Bratislava: ALFA, 1986.
- [20] ŠEFRÁNEK, J.: *Inteligencia ako výpočet.* Bratislava: IRIS, 2000.
- [21] ŠIMKOVIC, J., TEREK, M.: *Systémové modelovanie*. Bratislava: ALFA, 1987.
- [22] TURING, A.: *Computing machinery intelligence*. Mind 59, 1950.

- [23] UNČOVSKÝ, L.: *Operačná analýza v riadení podnikov*. Bratislava: ALFA, 1985.
- [24] ZÁVODNÝ, P., CHUDÁ, D.: *Globalizácia, nová ekonomika a podnikové informačné systémy*. In Ekonomika a informatika. Bratislava. 2006. – ISSN 1336-3514. – Roč. 4, č. 1 (2006), s. 75-91.
- [25] ZELINKA, I.: *Umělá intelligence v problémech globální optimalizace*. Praha: BEN-technická literatúra, 2002.

prof. Ing. Ivan Brezina, CSc.,
Katedra operačného výskumu a ekonometrie, Fakulta hospodárskej informatiky,
Ekonomická univerzita v Bratislave, Dolnozemská cesta 1, 852 35 Bratislava,
brezina@euba.sk, 0267295821

Ing. Zuzana Čičková, PhD.,
Katedra operačného výskumu a ekonometrie, Fakulta hospodárskej informatiky,
Ekonomická univerzita v Bratislave, Dolnozemská cesta 1, 852 35 Bratislava,
cickova@euba.sk, 0267295820

Návrh hybridného algoritmus umiestňovania dvojrozmerných útvarov na prepravnú plochu

Proposition of hybrid algorithm for solving two dimensional strip packing problem

Ján Pittner

Abstract:

Proposed article deals with proposition of hybrid algorithm designed to solve strip packing problem in two dimensional space. After mentioning the common algorithms, that are used to solve strip packing problem, we propose parallel hybrid algorithm, that reduces time to solve selected problems.Currently there are several existing heuristic algorithms, namely genetic algorithms, that are achieving very good results, however based on previous analysis, their weak point is usually in generating initial solutions, or any operation like mutation or crossover that works with deterministic algorithms. This problem together with genetic algorithm that we are currently working on was the main factor of creating this article. Although deterministic algorithms like those mentioned in this paper seem to be throughoutly mapped, in our opinion there is still place for additional research. Proposed algorithm is not perfect, it has it's flaws, however for purpose of our future genetic algorithm, it is suitable and useful.

Keywords:

object placing, hybrid algorithm, parallelization, optimalisation, two dimensional space

ACM classification: D1.3, G.0, G.1.0, G.4

Úvod

V súčasnej dobe otvorenej ekonomiky, keď sa na trhu vyskytuje veľký počet subjektov rastie potreba zefektívňovania jednotlivých procesov, pretože len efektívne nastavené procesy môžu zaručiť naplnenie cieľa, to jest maximalizácie zisku. Nesprávne nastavené procesy môžu naopak spôsobovať spomedzi mnohého aj zbytočné časové straty a tým pádom klesá aj objem zisku.

Na trhu tovarov sa spravidla nachádzajú tri kategórie subjetkov. Prvou kategóriu sú firmy(maloobchody) ponúkajúce výrobky zákazníkom, teda druhej kategórii. Firmy zväčša sprostredkúvajú predaj tovarov zákazníkom, pričom zisk dosahujú stanovením si určitej marže. Treťou kategóriou sú napokon výrobcovia tovarov, pričom spravidla figurujú ako dodávavatelia. Práca sa zaoberá dodávateľským reťazcom a jeho optimalizáciou, priblížme si teda tento proces.

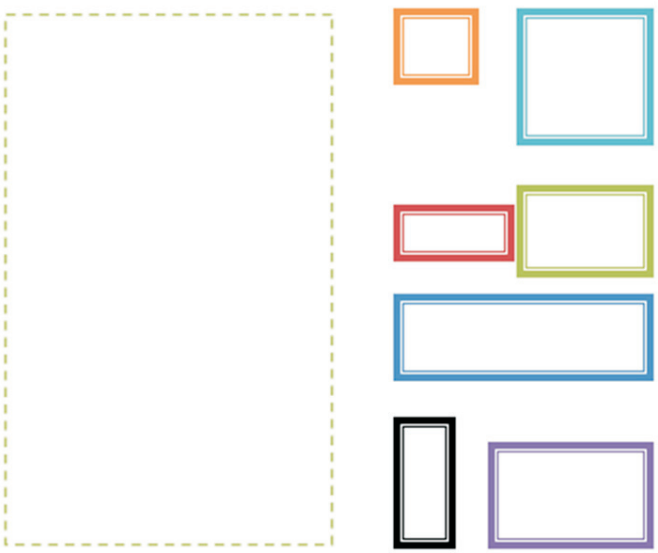
Dodávateľ je fyzická alebo právnická osoba, ktorá ma odberateľovi jednorázovo dodať, alebo mu dodáva za odplatu v dohodnutom čase alebo lehotách určitý výrobok. Zvyčajne sa nestáva, že by odberateľ zakúpil len jednu jednotku výrobku, respektíve výrobok, ktorý by sa skladal len z jednej časti.

Majme fast-food, ktorý od svojho dodávateľa odoberá potraviny a nápoje, ktoré predáva zákazníkom. Dodávka sa uskutočnuje raz mesačne, pričom potraviny sú uskladňované v chladiacich boxoch, aby vydržali dlhšie čerstvé. Fast-food má vysokú návštevnosť a preto aj dodávky bývajú zväčša objemnejšie. Požiadavkou odberateľa je teda včasná dodávka tovarov, aby nenastala situácia, v ktorej fast-food nie je schopný ponúknuť zákazníkom svoje služby a teda situácia ktorej dôsledkom je pokles zisku.

Dodávateľ na druhej strane síce z objednávky vie aké tovary bude musieť dodať, avšak pri maximalizácii zisku musí pozerať aj na prepravné náklady[1)]. Preto sa snaží pri dodávkach tovaru (či už priamo do predajnej prevádzky, alebo do skladu) pokryť čo najviac miesta v prepravnom priestore. Čím viac tovarov pojme jedna jednotka prepravného zariadenia, tým menej prepravných jednotiek musí expedovať, čím šetrí náklady na dopravu. Predpokladajme, že v tomto prípade dodávateľ využíva špedičnú kamiónovú spoločnosť a prenajíma si kontajnery, do ktorých jeho zamestnanci ukladajú tovary určené na expedíciu. Tovary sú balené z hygienických dôvodov v plastových foliách, ktoré sú uložené v kartónových boxoch.

Dodávateľ teda rieši problém optimalizácie prepravného miesta v kontajneroch (prepravných plochách). Problém vzniká vtedy, keď nie sú všetky boxy rovnakej veľkosti. V dôsledku krehkosti tovarov, predpokladajme, že boxy nemožno umiestňovať na seba, ale len vedla seba.

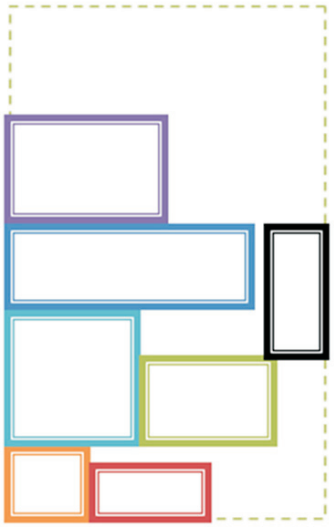
Máme teda prepravnú plochu a boxy, ktoré na ňu treba vhodne umiestniť:



Obrázok č. 1: *Prepravná plocha a boxy – zadanie*

Na obrázku 2 vidíme jedno z možných riešení. Samozrejme, aby sme vedeli ohodnotiť toto riešenie, naskytá sa otázka, akým algoritmom bolo vypočítané? Existuje mnoho variantov na výpočet tohto typu problému. Postupne prejdeme najznámejšie a spolu s grafickou ukážkou si osvojíme princípy ich fungovania.

1) Dodávky materiálu tvoria rozsiahlu časť logistiky, avšak v tejto práci sa zameriavame len na dodávku z miesta A do miesta B z hľadiska optimalizácie prepravného miesta.



Obrázok č. 2: *Prepravná plocha a boxy – možné riešenie*

Ako teda stanoviť vhodnosť jednotlivých algoritmov? V nasledujúcej kapitole si exaktne vysvetlíme fungovanie jednotlivých algoritmov, vyberieme tie, ktoré sú z hľadiska efektívnosti a času vhodné a pokúsime sa ich včleniť do hybridného algoritmu.

1. Definícia problému

Tento typ problému sa vo všeobecnosti nazýva problém balenia koša, alebo problém balenia pásu a je NP-úplným problémom. Formálna definícia hovorí, majme objekty rôznych objemov, ktoré musia byť zabalené do konečného počtu košov, ktoré majú určitú kapacitu tak, aby sme minimalizovali počet potrebných košov. Existujú mnohé variácie tohto problému ako napríklad 2D balenie, lineárne balenie, balenie podľa váhy, balenie podľa ceny atď. Praktické využitie je potom napríklad plnenie kontajnerov, nakladanie kamiónov obmedzených váhovou kapacitou, rozmiestňovanie technológií na mikroprocesor atď.

Z predchádzajúceho výkladu teda môžeme očakávať, že máme zadanú ohraničenú oblasť O a množinu útvarov S_i kde $i=1,2,3....n$ pričom n je dostatočne veľké. Úlohou je potom rozmiestniť útvary S_i tak, aby bola minimalizovaná plocha, potrebná na ich umiestnenie.

2. Algoritmy riešiace problém balenia pásu

Problém hľadania optimálneho riešenia pre problém balenia pásu je NP-úplný, tým pádom sa výskum zameriava hlavne na vývoj aproximačných algoritmov. Aproximačné algoritmy nájdu takmer optimálne riešenia, ale nezaručujú nájdenie optima pre každú skupinu dát.

Aproximačné, alebo aj heuristické algoritmy však implicitne využívajú deterministické algoritmy pre potreby generovania inicializačných riešení, resp. pri operáciách ako je mutácia alebo prekríženie. Pri analýze existujúcich aproximačných algoritmov sme zistili, že hoci v súčasnosti existujú takmer dokonalé algoritmy, mnohé z nich sa sústredia na nastavovanie evolučných parametrov, vhodné metódy selekcie a opomíňajú základné algoritmy, ktoré predstavujú podstatnú časť záťaže procesora.[2)] Motiváciou tohto článku bola práve táto oblasť, pretože optimalizácia týchto fundamentálnych princípov spôsobí podľa nášho názoru nárast výkonnosti aproximačného algoritmu.

2) KRAUSPE, Kamil. 2011. An evolutionary algorithm for mixed postman problem. In *Mezinárodní Baťova konference pro doktorandy a mladé vědecké pracovníky : [recenzovaný zborník] : 7. ročník : 12. dubna 2011* [elektronický zdroj]. – Zlín : Univerzita Tomáše Bati ve Zlíně, 2011. ISBN 978-80-7454-013-4, s. [1-11].

Väčšina algoritmov funguje tak, že umiestňuje útvary na pás využívajúc pri tom jeden z piatich prístupov.

- algoritmus "Dolu vlavo"

  Tento typ algoritmu sa pri hľadaní pozície pre útvary snaží nastaviť pozíciu čo najbližšie spodku kontajnera a potom čo najviac naľavo, bez toho aby súčasný útvar prekryl už uložený útvar. Neplatí tu podmienka zoradenia útvarov pred ukladaním.

- úrovňovo orientované algoritmy

  Algoritmus začína zoradením útvarov podľa výšky (od najvyššieho po najnižší). Následne sa uskutočnuje balenie na rôznych úrovniach, ktoré určujú pozíciu spodku útvaru. Prvú úroveň predstavuje spodok kontajnera a nasledovné úrovňe sú definované výškou najvyššieho útvaru, ktorý bol umiestnený na predchádzajúcej úrovni.

- algoritmus delenia

  Pre vysvetlenie uvedieme porovnanie s úrovňovo orientovanými algoritmami. Úrovňovo orientované algoritmy sú v podstate podmnožinou algoritmov delenia, keďže delia kontajner horizontálne na bloky určitej šírky. Pod pojmom algoritmus delenia máme na mysli potom algoritmus, ktorý rozdelí vertikálne kontajner na menšie časti, v závislosti od šírky obdĺžnikov. Algoritmus vždy začína zoradením útvarov podľa šírky pred ukladaním.

- algoritmus regálov

  Tieto algoritmy sú modifikáciou úrovňovo orientovaných algoritmov, ktoré sa vyhýbajú problému zoradenia zoznamu útvarov pred ich ukladaním. Pri stanovovaní nasledovnej úrovňe však narozdiel od úrovňovo orientovaných algoritmov, kde je spodok nasledovnej úrovne určený výškou najvyššieho útvaru na súčasnej úrovni, sa pri algoritme regálov stanovuje ich výška parametrom *r,* pre ktorý platí :

  $$0 < r < 1$$

- hybridné algoritmy

  Tieto špeciálne typy algoritmov, využívajú dva alebo viac typov algoritmov popísaných vyššie v kombinácii s heuristickými algoritmami. Môžu, ale nemusia zahŕňať zoradovanie útvarov pred samotným ukladaním.

2.1 Podriadené algoritmy (The Slave Algorithm)

Podriadený algoritmus je prístup využívaný na rozhodnutie, do ktorého regálu sa má útvar umiestniť v prípade, že boli všetky regály stanovené.

Pre úrovňovo orientované algoritmy, ktoré som si v projekte zvolil ďalej skúmať, má každá úroveň (teda regál) v koši rovnakú šírku *C.* Tým pádom sa po vypočítaní úrovne stáva rozhodnutie typom jedno-rozmerného plnenia nádoby. To znamená, že počet používaných úrovní zodpovedá počtu jednorozmernej nádoby kapacity *C.* Ďalej je potom tento jednorozmerný problém riešený podriadenými algoritmami. Existuje množstvo jednorozmerných algoritmov, ktoré môžu byť využité ako podriadené algoritmy. Ako príklad zvolme algoritmy Next-fit a First-fit. Všetky algoritmy majú svoje výhody a nevýhody.

Porovnajme teda tieto dva algoritmy tým, že vysvetlíme na akom princípe pracujú.

Next-fit algoritmus umiestňuje každý útvar na najvyššiu úroveň na ktorú sa zmestí. Naproti tomu first-fit algoritmus umiestni každý útvar na najnižšiu úroveň na ktorú sa zmestí. Z hľadiska úspornosti je najčastejšou voľbou práve First-fit algoritmus, pretože umožnuje umiestniť útvary na všetkých úrovniach s cieľom čo najnižšieho umiestnenia, zatiaľ čo next-fit iba umožnuje využiť súčasnú a teda najvyššiu úroveň, pričom spätné umiestňovanie (napríklad pri zistení miesta

pre súčasný útvar o 3 úrovne nižšie) nieje možné. Čo sa týka prípadov použitia je first-fit algoritmus vhodným algoritmom pre umiestňovanie pokiaľ je možné umiestniť všetky útvary do kontajnera pred vykonaním určitej akcie.

Next-fit algoritmus by bol potrebný v prípade, že akcia (napr. práca) začala a súčasne musíme umiestnovať do nádoby ďalšie útvary. Nech je výška nádoby čas a útvary trvania určitých prác, plynutím času sa pomyselný kontajner „zmenšuje“ čiže už nie sme schopní vrátiť sa v čase a naplánovať práce inak, pretože určitý čas už uplynul.

2.2 On-line a off-line algoritmy

Pri zvolení si správneho algoritmu je rozhodujúcim faktorom taktiež to, či musia byť jednotlivé útvary pred umiestnením zoradené. Majme situáciu, v ktorej útvary prichádzajú jeden po druhom. Pri príchode každého útvaru, musíme tomuto útvaru priradiť miesto v nádobe. Až po tejto akcii zistíme parametre nasledujúceho útvaru.

Tomuto typu prostredia hovoríme on-line prostredie. Tým pádom všetky on-line algoritmy musia predpokladať, že nebudú mať žiadnu znalosť o nasledujúcich útvaroch, takže zoradenie pred umiestnovaním nepripadá do úvahy.

Naproti tomu v off-line prostredí je možné zobraziť celý zoznam útvarov dopredu a ľubovoľne ich zoradiť do akéhokoľvek poradia predtým, než sú umiestnené do nádoby. Off-line algoritmy teda predpokladajú znalosť celého problému pred vykonávaním umiestňovania.

On-line algoritmus býva využívaný v situáciách kde sa práce musia vykonávať v určitom poradí. Napríklad jednotlivé predmety môžu byť predmetom rôznych priorít alebo termínov.

Off-line algoritmus býva potom využívaný v situáciách kde na poradí nezáleží. Ako príklad uveďme výrobu kusu oblečenia. Prípad, že z kusu materiálu vystrihneme ako prvý rukáv a až potom golier nemá význam, pretože na poradí nezáleží a až po vystrihnutí všetkých častí môžeme zostrojiť tričko.

Jedným z ďalších faktorov, ktoré môžu zavážiť pri výbere algoritmu je potrebné množstvo pamäte. Off-line algoritmus v tomto prípade zaberá viac pamäťových prostriedkov, keďže pred samotným umiestňovaním triedi.

2.3 Úrovňové a regálové algoritmy

Priblížme si teraz tieto dve skupiny algoritmov, ich funkcionalitu a názornú ukážku. Ako prvé si predstavíme niektoré regálové algoritmy. Regálové algoritmy sú typom úrovňových algoritmov, ktoré nezoraďujú útvary. Preto narozdiel od úrovňových algoritmov sú regálové algoritmy radené do triedy on-line algoritmov. Úrovne sú teda určované nie najvyšším útvarom ale fixnými „regálmi“, ktorých výška je určená parametrom r. Každý nový útvar je potom klasifikovaný podľa jeho výšky. Samotné balenie prebieha ako konštrukcia skupiny „regálov“, pričom útvary podobných výšok sú ukladané na rovnaký regál. Veľkosť regálov má potom tvar r^k, každý útvar má určitú výšku h a je umiestnený do regálu, ktorý má výšku r^k. Tým pádom platí rovnica

$$r^{k+1} \leq h = r^k$$

$$\text{kde } k \in Z$$

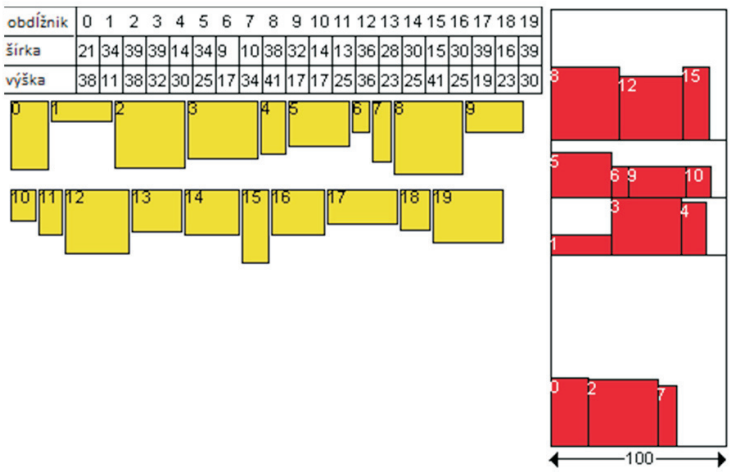
Parameter r je známy vopred a jeho cieľom je obmedziť počet nevyužitého miesta povoleného na každom regáli. Vysvetlime si, ako nastavovanie parametra r ovplyvňuje balenie.

V prípade, že parameter r nadobúda malú hodnotu ($r \approx 0$) je rozsah výšok povolených na každom regáli veľky. Často sa stretávame so situáciou keď máme len malé množstvo útvarov na umiestnenie a nechceme, aby sa na jednotlivých regáloch nachádzal len jeden útvar, s tým, že

regále by zostali neobsadené. Všeobecne pre malé *r* platí, že máme na výber menej možností výberu výšok regálov, pričom každý regál obsahuje útvary s väčším rozsahom výšok.

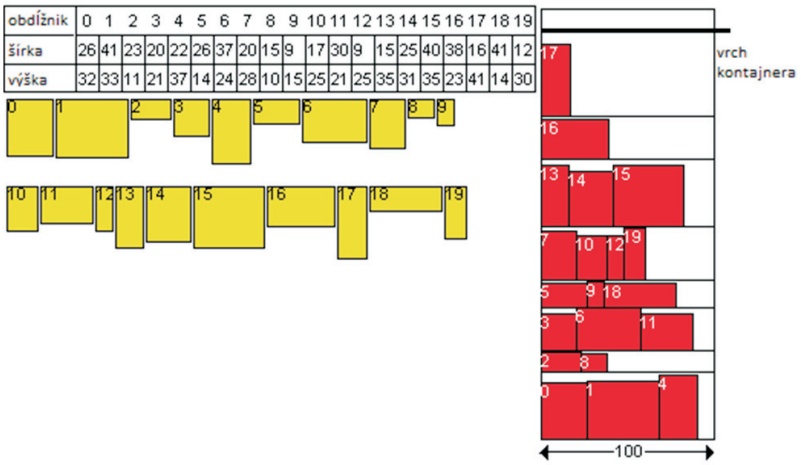
Naopak v prípade, že parameter *r* nadobúda veľkú hodnotu (*r≈1)* bude každý z regálov obsahovat len útvary veľmi podobné svojou výškou. Využitie tohto prípadu je užitočné, v prípade, že útvary v zozname majú veľmi podobné výšky, čize algoritmus musí rozlišovať aj menšie rozdiely vo výškach. Všeobecne platí, že vysoká hodnota parametra *r* spôsobí širší výber výšok regálov s útvarmi veľmi podobných výšok.

O výsledkoch riešenia v závislosti od parametra pojednávajú nasledujúce obrázky



Obrázok č. 3: *Výsledok regálového Next-fit algoritmu s parametrom r=0,3*[3)]

Ako môžeme vidieť z obrázku 3, máme 20 útvarov so stanovenou šírkou a výškou. Regálový next-fit algoritmus bude vysvetlený v nasledovnej kapitole, avšak za povšimnutie tu stoja hranice regálov. Tie boli stanovené vďaka parametru *r* tak, že zobrazená časť nádoby nebola schopná generovať dostatočný počet regálov. Pri ešte nižšej hodnote parametra *r* očakávame ešte benevolentejšie hranice regálov. Je teda zrejmé, že parameter *r* je nutné nastavovať aj podľa toho, ako sme limitovaný výškou kontajneru.



Obrázok č. 4: *Výsledok regálového Next-fit algoritmu s parametrom r=0,8*[4)]

3) Zdroj obrázku: výstup php algoritmu naprogramovaného podľa http://users.cs.cf.ac.uk

4) Zdroj obrázku: výstup php algoritmu naprogramovaného podľa http://users.cs.cf.ac.uk

Zmena parametra *r* z 0,3 na 0,8 je zrejmá už len z pohľadu na obrázky 3 a 4. Hranice sú oveľa striktnejšie a taktiež sa na jednotlivých regáloch nachádza väčší počet útvarov.

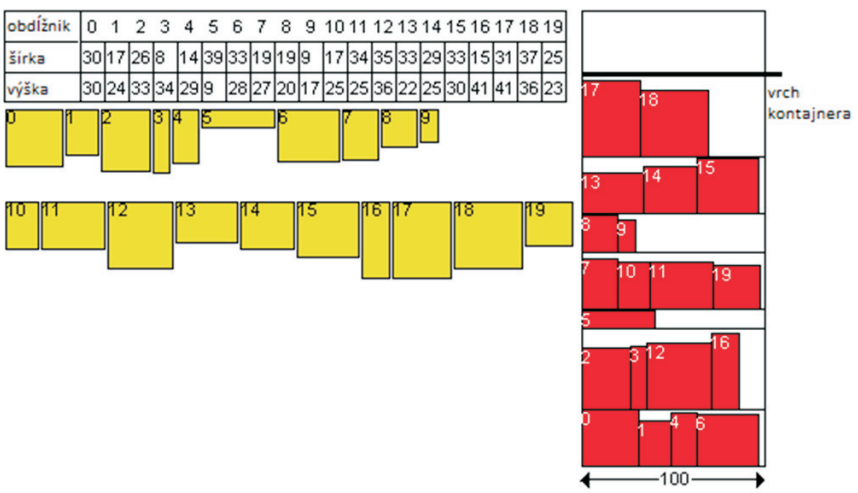
Poďme si teraz priblížiť niektoré algoritmy nachádzajúce sa v skupine regálových algoritmov.

2.3.1 Regálový Next-fit algoritmus

Tento typ algoritmu sa pri balení každého útvaru snaží zvoliť jeho pozíciu najviac vľavo, ako je to možné na najvyššom regáli. V prípade, že na danom regáli nieje miesto, alebo regál s takouto výškou neexistuje je vytvorený nový regál, ktorý sa stáva aktívnym pre danú výšku. Regále rovnakej výšky nachádzajúce sa pod aktívnym regálom nie sú zohľadňované. Ako vidíme napríklad z obrázku 4, pre útvar číslo 3 bol vytvorený nový regál, pretože jeho umiestnenie do druhého regálu nepripadalo kvôli výškovému obmedzeniu v úvahu[5)].

2.3.2 Regálový First-fit algoritmus

Tento algoritmus funguje tak, že je vybraný najspodnejší regál korektnej výšky, na ktorý je možné daný útvar umiestniť. Ak nemáme regál požadovanej výšky, alebo žiadny z regálov nespĺňa požiadavku na miesto, vytvorí sa nový regál s výškou podobnou danému obdĺžniku.



Obrázok č. 5: *Regálový First-fit algoritmus s parametrom r=0,7*[6)]

Ako vidíme z obrázka 5 už pre druhý útvar nezodpovedala výška prvého regálu (max. 30, útvar 2 má 33) a preto bol vytvorený nový regál.

Úrovňové algoritmy sú narozdiel od regálových off-line algoritmy. Z predchádzajúcich definícií teda vieme, že off-line algoritmy zahŕňajú znalosť rozmerov všetkých útvarov a ich následné zoraďovanie podla výšky zostupne. Prvý útvar, ktorý bude balený má teda najvyššiu výšku a posledný najnižšiu. Balenie sa potom uskutočnuje na postupnosti úrovní. Každý útvar je postupne balený do kontajnera umiestnením spodnej hrany tak, že kopíruje hranicu jednej z úrovní. Prvá úroveň je teda spodok kontajnera a každá nová úroveň je potom definovaná horizontálnou priamkou rovnobežnou a prechádzajúcou vrcholom najvyššieho útvaru predchádzajúcej úrovne.

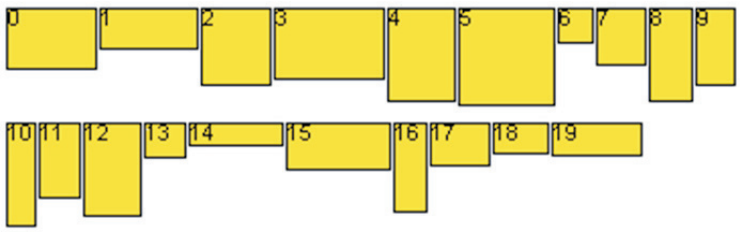
5) V tomto prípade zohral najväčšiu úlohu parameter r, ktorý bol nastavený na vysokú hodnotu. Pri pohľade na obrázok 4 je zrejmé, že situácia sa radikálne mení pri zmene parametra *r*.

6) Zdroj obrázku: výstup php algoritmu naprogramovaného podľa http://users.cs.cf.ac.uk

2.3.3 Úrovňový algoritmus next-fit so znižujúcou sa výškou

Tento algoritmus je úrovňový algoritmus, ktorý využíva prístup next-fit na balenie zoradeného zoznamu útvarov. Útvary sú postupne balené tak, že sú zarovnané zľava na jednej úrovni, až pokial sa nasledujúci útvar na danú úroveň nezmestí. Tento útvar je potom využitý na stanovenie novej úrovne a balenie pokračuje na tejto úrovni. Predchádzajúce úrovne dalej nie sú zohľadnované.

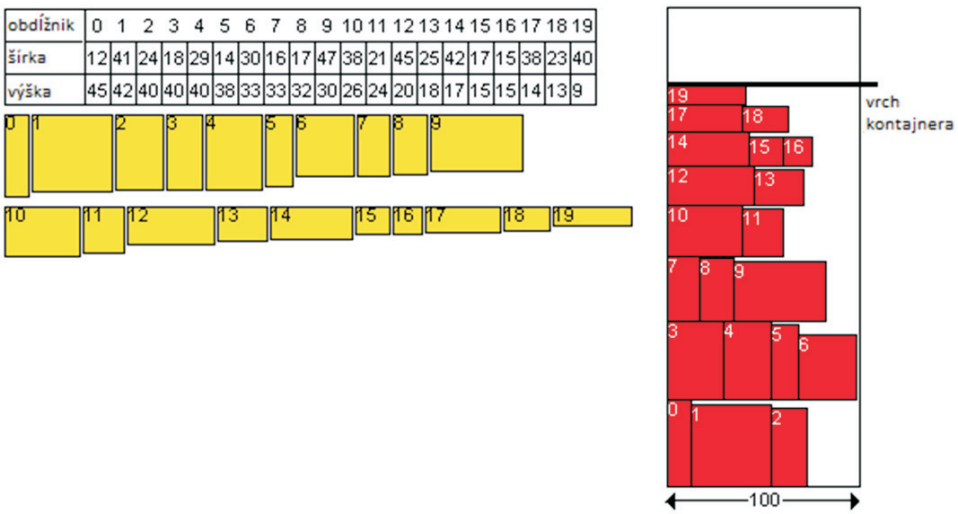
obdĺžnik	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
šírka	38	42	30	47	29	41	15	21	18	16	12	17	24	17	40	45	14	25	23	38
výška	26	17	33	30	40	42	15	24	40	33	45	32	40	15	9	20	38	18	13	14



Obrázok č. 6: *Vygenerované nezoradené útvary*[7)]

Po vygenerovaní útvarov nasleduje ich zoradenie podľa výšky a následné ukladanie pomocou algoritmu.

obdĺžnik	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
šírka	12	41	24	18	29	14	30	16	17	47	38	21	45	25	42	17	15	38	23	40
výška	45	42	40	40	40	38	33	33	32	30	26	24	20	18	17	15	15	14	13	9



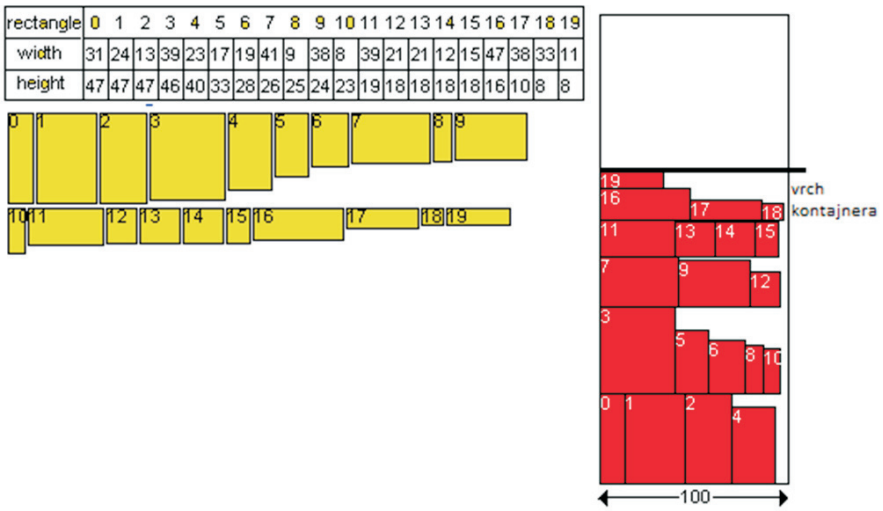
Obrázok č. 7: *Zoradenie a uloženie pomocou algoritmu next-fit*[8)]

2.3.4 Úrovňový algoritmus First-fit so znižujúcou sa výškou

Úrovňový algoritmus First-fit so znižujúcou sa výškou pracuje tak, že každý útvar je umiestnený na prvú úroveň kam sa zmestí. V prípade, že sa nezmestí do žiadnej z existujúcich úrovní, vytvorí sa nová úroveň. Rozdiel oproti Next-fit algoritmu je teda práve v tom, že útvar je umiestnený na prvé miesto kde sa zmestí, to znamená že sa testuje či sa na predchádzajúcich úroveniach nachádza plocha volná na jeho umiestnenie. Toto môže znamenať veľké úspory miesta v kontajneri.

7) Zdroj obrázku: výstup php algoritmu naprogramovaného podľa http://users.cs.cf.ac.uk

8) Zdroj obrázku: výstup php algoritmu naprogramovaného podľa http://users.cs.cf.ac.uk



Obrázok č. 8: *Umiestnenie zoradených útvarov do kontajnera úrovňovým algoritmom First-fit*[9)]

2.3.5 Ostatné typy algoritmov geometrickej optimalizácie 2D priestoru

Hore uvedené typy algoritmov zďaleka nie sú všetky, ktoré má potencionálny riešiteľ k dispozícii, avšak detailné rozpísanie všetkých z nich by presahovalo rozsah práce. Za spomenutie určite stojí inicializačná fáza kombinatorického algoritmu Benderových rezov[10)], algoritmus Akeba H.[11)], ktorý uplatňuje prístup hľadania a spätných krokov či heuristika SPEA2[12)] v kombinácii so štandardnými algoritmami v evolučnom algoritme umiestňovania útvarov na 2D pás od Colheo, Wannera, Souza a Carrana.

Práve z kombinácie vhodných vyššie spomenutých algoritmov sa skladá náš hybridný paralelný algoritmus.

3. Hybridný paralelný algoritmus balenia pásu

V súčasnosti, keď viacjadrové počítače predstavujú štandard prichádza potreba vytvárať algoritmy využívajúce túto hardvérovú kapacitu. Napriek obtiažnostiam pri paralelizácii môžeme pozorovať podstatný nárast výkonu, čo môže spôsobiť, že dané problémy sú riešené omnoho rýchlejšie. Druhou výhodou je to, že vznikajú rôzne hybridné algoritmy, ktoré sú niekedy efektívnejšie čo sa týka počtu krokov smerujúcich k vyriešeniu úlohy ako pôvodné algoritmy.

Pri návrhu paralelného algoritmu bola najdôležitejšia otázka, ako vhodne využiť výpočtovú kapacitu viacerých jadier. Museli sme teda odpovedať na otázku, ako je problém možné efektne deterministicky rozdeliť. Ak bližšie preskúmame obrázky jednotlivých algoritmov, možeme konštatovať nasledovné: jediným prvkom, o ktorom neviem dopredu povedať akú bude mať

9) Umiestnovaniu predchádzalo zoraďovanie, čo sme však z hladiska šetrenia miesta nezahrnuli do práce ako obrázok. Zdroj obrázku: výstup php algoritmu naprogramovaného podľa http://users.cs.cf.ac.uk

10) Cote, J., Dell'Amico, M., Iori, M.: Combinatorial Benders' Cuts for the Strip Packing Problem [online], 20.4.2013 https://www.cirrelt.ca/DocumentsTravail/CIRRELT-2013-27.pdf

11) Akeb, H., Hifi, M., Lazure, D.: An improved algorithm for the strip packing problem. In *Computer Science and Information Systems (FeDCSIS)*, str. 357 – 364, ISBN:9781467307086

12) Coelho, D.G., Wanner E.F., Souza S.R., Carrano, E.G: A multiobjective evolutionary algorithm for the 2D Guillotine Strip Packing Problem, In *Evolutionary Computation (CEC)*, str. 1-8, ISBN:9781467315104

veľkosť je ušetrená plocha. Tento prvok je teda nedeterministický. Pri našom nasledovnom postupe s ním teda tak budeme uvažovať.

Filozofiou nášho algoritmu je teda algoritmus navrhnúť tak, aby každé jedno jadro spracovávalo deterministický algoritmus. Tým pádom sme však obmedzený na riešenie pomerne špecifických prípadov.

1. Algoritmus predpokladá využitie 100 % možnej kapacity pásu. Táto podmienka nám teda dopredu hovorí, že efektívnosť algoritmu bude v rýchlosti spracovania a nie v optimalizácii ušetreného miesta
2. V súvislosti s bodom jedna potom uvažujeme o rozmiestňovaní útvarov, ktorých celková plocha je vyššia ako celková plocha pásu. Teda vzniká situácia, že aj v prípade optimálneho riešenia nebude možné uložiť na pás všetky útvary.
3. Predpokladáme, že ukladané útvary majú tvar obdĺžnika resp. štvorca.

Náš hybridný algoritmus kombinuje dva klasické prístupy; regálový a úrovňový algoritmus. Algoritmy boli vybrané z hľadiska úspory času a ich efektívnosti. Parameter *r* je asi najdôležitejšou časťou algoritmu, pretože jeho spôsoby výpočtu môžu zefektívniť, ale taktiež aj zhoršiť kvalitu algoritmu.

3.1 Konštrukcia algoritmu

Činnosť algoritmu možno zhrnúť nasledovne:

1. Zisti počet jadier procesora.
2. Náhodne rozdeľ balené útvary do skupín, ktorých počet je zhodný s počtom jadier procesora.
3. Rozdel pás na rovnako veľké úseky, ktorých počet sa rovná počtu jadier procesora.
4. Pomocou metódy *sort()* zoraď útvary v jednotlivých skupinách od najväčšieho po najmenší.
5. V prípade, že ľubovolný najväčší útvar je vyšší ako je výška ľubovolného úseku, zlúč dva úseky do jedného a k nim príslušné skupiny do jednej. Tieto skupiny znovu zoraď pomocou metódy *sort()*.
6. V paralelnom cykle *Paralel.For* vykonaj pre každé jadro[13] deterministický *first-fit* algoritmus.
7. Po ukončení behu algoritmu na všetkých jadrách ulož zvyšné útvary do poľa a zoraď toto pole vzostupne.
8. Pre každú časť pása vykonaj deterministický *first-fit* a ulož útvary, ktoré je možné uložiť.

Výhodou takéhoto algoritmu z pohľadu teórie je dĺžka exekúcie. Ak predpokladáme optimálnu situáciu a teda, že sa využijú všetky jadrá, očakávame zrýchlenie v priemere o 15 % až 50 % na jedno jadro procesora.

Na testovanie algoritmu sme si vytvorili generátor útvarov s náhodnými veľkosťami, ktoré však neprekročia 20 % z plochy pásu[14]. Veľkosť pásu je generátoru zasielaná ako parameter.

3.2 Dosiahnuté výsledky

Dosiahnuté výsledky sme zhrnuli do tabuľkovej formy doplnenej o graf. Každý z algoritmov bol kvôli odstráneniu odchýliek spustený 500 krát. Napriek tomu treba uvažovať pri interpretácii

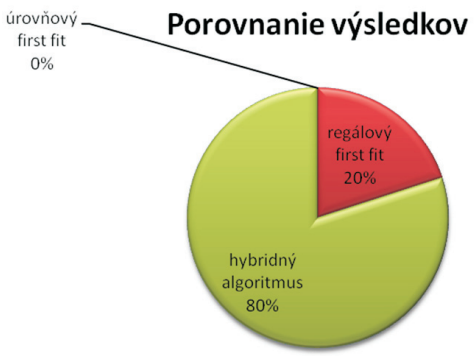
13) Ktorých počet zistíme ako počet častí pásu

14) V prípade, že by veľkosť jedného útvaru prekročila 20 % dosahoval by v prípade, že počítač na ktorom algoritmus beží má viac než 8 jadier suboptimálne výsledky.

aj s miernou odchýlkou. Tabuľka č. 1 nám ukazuje priemerný čas behu algoritmu s rozličným počtom ukladaných útvarov pre rôzne veľkosti pásu.

Tabuľka č. 1 *Dosiahnuté výsledky*[15)]

veľkosť pásu	počet útvarov	priemerná veľkosť útvaru	exekučný čas úrovňového first fit	exekučný čas regálového first fit	exekučný čas nášho algoritmu
550	250	5,8146	20,12	25,9	14,7
600	280	6,128	24,128	30,844	20,12
700	350	8,55	27,6	32,8	22,11
800	400	7,65	31,5	37,77	26,24
850	420	6,782	33,97	39,43	29,56
900	450	5,97	36,71	42,11	31,5
1000	570	7,138	39,97	45,49	34,27
1050	720	6,71	42,45	48,16	37,71
2000	1200	8,12	75,12	94,58	76
4000	2500	7,54	162,44	164,844	170,15
10000	5000	4,54	380,15	394,74	450,12



Graf č. 1 *Porovnanie výsledkov*[16)]

4. Diskusia a návrhy na zlepšenie

Nami navrhnutý hybdriný algoritmus dosiahol v 80 % testovaných prípadov lepšie výsledky ako dva konkurenčné algoritmy. Algoritmus teda potvrdil hypotézu, že paralelizácia prinesie isté zlepšenie v otázke exekučného času. Algoritmus však pracoval pod pomerne špecifickými podmienkami, ktoré významne obmedzujú jeho využitie v praxi.

Algoritmus bol naprogramovaný v jazyku ?#, ktorý má značnú podporu paralelného programovania. Naproti tomu konkurenčné algoritmy boli spracované v jazyku PHP. Rozdiel v rýchlosti týchto dvoch jazykov pravdepodobne penalizoval riešenie v jazyku PHP. Odborná literatúra uvádza, že jazyk ?# je zhruba o 15 %-20 % rýchlejší ako jazyk PHP.

Ako sa počas testovania ukázalo, náš algoritmus má tendenciu zhoršovať svoje exekučné časy vzhľadom na zvyšujúcu sa veľkosť pásu a množstva ukladaných útvarov. Toto je možné vysvetliť veľmi jednoducho. Totiž algoritmus po tom, ako sa všetky parciálne pásy naplnili uloží do pola neumiestnené predmety, zoradí ich a snaží sa vložiť do jednotlivých parciálnych pásov

15) Výsledky sú uvádzané v sekundách. Vlastný zdroj.

16) Vlastný zdroj

možné útvary. S rastúcim počtom útvarov potom rastie aj časové kvantum potrebné na zoradenie a pokusné vkladanie do častí pásu. Z toho možno usudzovať, že čím viac bude ukladaných útvarov, tým horšie časy bude algoritmus vykazovať.

Záver

Paralelizácia je jednoznačným prínosom pre počítanie úloh operačného výskumu. Otázka, ktorú sa snaží mnoho ľudí riešiť je jej vhodná implementácia. Paralelizácia nevhodných častí kódu môže potom spôsobiť, že čas behu sa namiesto očakávaného zníženia zvýši a preto je vhodné algoritmus ako celok analyzovať.

Medzi základné typy paralelizácie patrí paralelizácia jednoduchých činností, hlavne teda cyklov. Náš algoritmus v podstate rieši namiesto jednej úlohy *n*-rovnakých úloh, kde *n* predstavuje počet jadier procesora. Toto nám v tomto konkrétnom prípade zabezpečí maximálne možné vyťaženie procesora, pretože sú v ideálnom prípade využívané všetky jadrá. Efektívnejšia paralelizácia by bola taká, kde by sa na jednotlivé vlákna delegovali špecifické úlohy jedného algoritmu. Ak je však algoritmus rekurzívny, teda treba napríklad zoradiť útvary pred ich samotným uložením, je paralelizácia prakticky nemožná.

Preto by do budúcna bolo vhodné zamyslieť sa aj nad samotnými matematickými krokmi daného algoritmu a pokúsiť sa ich upraviť tak, aby boli paralelizovateľné. V našom hybridnom algoritme sa to čiastočne podarilo, avšak za cenu niektorých obmedzení.

Cieľom navrhnutého algoritmus nebolo konkurovať heuristikým algoritmom, ktoré riešia rozličné variácie problému balenia pásu. Navrhnutý algoritmus by mal znamenať prínos pre niektoré heuristické algoritmy (v závislosti od ich spôsobu generovania a modifikácie riešení) a v budúcnosti bude zohrávať práve túto úlohu v genetickom algoritme, ktorý v súčasnosti pripravujeme.

Použitá literatúra:

- [1] MAURICIO G. C. R., PINHO DE SOUSA, J.: *Metaheuristics: Computer Decision Making Applied Optimization.* Springer, 2003. ISBN 9781402076534.
- [2] ANDONOV, R. et al.: *Optimal semi-oblique tiling.* New York : ACM, 2001. ISBN1581134096.
- [3] MURPHEZ, R., PARDALOS, P.: *Cooperative Control and Optimization.* [s.l.] : Springer, 2002. ISBN1402005490.
- [4] RIBEIRO, C., MARTINS, S.: *Experimental and Efficient Algorithms.* Berlin : Springer, 2004. ISBN3540220674.
- [5] ROBINSON, S. et al.: *C# Programujeme profesionálně.* Brno : Computer Press, 2003. ISBN8025100855.
- [6] YANG, X., TEO, K.,CACCETTA, L.: *Optimization methods and applications.* Dordrecht : Springer, 2001. ISBN9780792368663.
- [7] DUPAĽ A., BREZINA I.: Logistika v manažmente podniku Bratislava : Sprint vfra, 2006. – 326 s. : grafy, obr., sch., tab. – ISBN 80-89085-38-5
- [8] BREZINA I., ČIČKOVÁ Z., REIFF M.: Kvantitatívne metódy na podporu logistických procesov – Bratislava : Vydavateľstvo EKONÓM, 2009. – 191 s. – ISBN 978-80-225-2648-7
- [9] GRELL, M., HYRÁNEK E.: Maticové modely na meranie výkonnosti produkčných systémov. In *E + M. Ekonomie a management : vědecký ekonomický časopis*. – Liberec : Ekonomická fakulta Technické univerzity v Liberci, 2012. ISSN 1212-3609, 2012, roč. 15, č. 1, s. 73-87. VEGA 1/165/08.
- [10] CHOCHOLATÁ M.: Lineárne programovanie pre manažérov, Bratislava : Vydavateľstvo EKONÓM, 2013. – 234 s. [14,021 AH]. – ISBN 978-80-225-3562-5

- [11] KRAUSPE, K.: An evolutionary algorithm for mixed postman problem. In *Mezinárodní Baťova konference pro doktorandy a mladé vědecké pracovníky : [recenzovaný zborník] : 7. ročník : 12. dubna 2011* [elektronický zdroj]. – Zlín : Univerzita Tomáše Bati ve Zlíně, 2011. ISBN 978-80-7454-013-4, s. [1-11].
- [12] BREZINA I.JR., ČIČKOVÁ Z.: *Solving the Travelling Salesman Problem Using the Ant Colony Optimization,* Management Information Systems, vol. 6, pp. 10-14, 2011.
- [13] COTE, J., DELL'AMICO, M., IORI, M.: *Combinatorial Benders' Cuts for the Strip Packing Problem* [online], 20.4.2013 https://www.cirrelt.ca/DocumentsTravail/CIRRELT-2013-27.pdf
- [14] AKEB, H., HIFI, M., LAZURE, D.: An improved algorithm for the strip packing problem. In *Computer Science and Information Systems(FeDCSIS),* str. 357 – 364, ISBN:9781467307086
- [15] COELHO, D.G., WANNER E.F., SOUZA S.R., CARRANO, E.G.: A multiobjective evolutionary algorithm for the 2D Guillotine Strip Packing Problem, In *Evolutionary Computation (CEC),* str. 1-8, ISBN:9781467315104
- [16] *Distributed & Scientific Computing, Cardiff University* [online], 1.4.2012 http://www.cs.cf.ac.uk/research/dsc.

Ing. Ján Pittner, PhD.
Katedra aplikovanej informatiky
Fakulta hospodárskej informatiky, Ekonomická univerzita v Bratislave,
Dolnozemská cesta 1, 852 35 Bratislava,
Telefón: 0903 630 151
e-mail: janci.pittner@gmail.com

Rôzne prístupy k analýze portfólia

Various Approaches to the Portfolio Analyses

Ivan Brezina jr.

Abstract

In this paper the author describe various possibilities, how to select the financial portfolio. The actuality of this these confirm the financial markets too. There are special rules on the markets and the economists try to understand the relationships for a long time. In the front of this paper are the historical methods, which were used to the portfolio selection few years ago and always are the ground of new methods, which deal in the portfolio selection today. The main topic is in summarization of the most used rates of risk in praxis.

Key words:

portfolio, portfolio selection, Value at Risk (VaR), Capital Asset Pricing Model (CAPM), Mean Absolute Deviation (MAD), Conditional Value at Risk (CVaR)

ACM classification: G.0, G1.6, G.3

Úvod

Pod pojmom portfólio sa zvyčajne označuje súbor komodít, ktorý bol vytvorený na základe zníženia rizika súvisiaceho s ľudskou činnosťou. Aj keď budeme pojmom portfólio označovať finančné portfólio, už na začiatku treba upozorniť, že je to iba časť problematiky tvorby a spravovania portfólia, ktorá môže byť spojené s rôznymi hospodárskymi činnosťami, ako napríklad: ekológia, energetika, strojárenstvo a iné.

Predmetom nášho skúmania bude finančné portfólio, ktoré chápeme ako súbor finančných aktív, ktoré sú investorom držané za účelom minimalizácie finančného rizika spojeného s investovaním. Finančné riziko je možnosť, že nepredvídané a nepredvídateľné udalosti budú mať za následok finančnú stratu.[1)] Práve na finančnú stratu, ktorá vyplýva z neznalosti sa zameralo už viacero ekonómov.

Ako prelomový sa označuje prístup, ktorý navrhol a bližšie sa mu venoval americký ekonóm Markowitz, ktorého dielo vydané v The Journal of Finance (1952) odštartovalo skúmanie možností diverzifikácie rizika. Ďalšie Markowitzovo dielo zaoberajúce sa touto problematikou Portfolio Selection – Efficient Diversification of Investment prehlbuje znalosti o investovaní na finančných trhoch. Práve v týchto dielach navrhol Markowitz svoj prístup k riadeniu finančného portfólia na základe metódy priemer- rozptyl. Za túto svoju prácu získal Markowitz spolu so Sharpem a Millerom roku 1990 Nobelovu cenu za ekonómiu.

1) Zenios, S. A.: Practical financial optimization, Decision Mmaking for Financial Engineers. Padstow: Blackwell Publishing. 2007.

1. Markowitzova formulácia riadenia rizika

Markowitz navrhol ako prvý model riadenia portfólia, vychádza pritom z historických cien aktív, pričom minimalizuje rozptyl portfólia pri danom očakávanom výnose. Takáto optimalizácia portfólia sa tiež často nazýva optimalizácia v priestore priemer- rozptyl. Výraz priemer sa tu pritom chápe ako priemerný, alebo očakávaný výnos investície a rozptyl je mierou rizika spojeného so správou portfólia. Matematický problém môže byť formulovaný rôznymi spôsobmi, ale problém ako taký sa dá všeobecne zhrnúť nasledovne[2)]:

1. Minimalizácia rizika pre špecifikovaný výnos;
2. Maximalizácia očakávaného výnosu pre špecifikované riziko;
3. Minimalizácia rizika a maximalizácia očakávaného výnosu, pokiaľ je použitý špecifikovaný faktor averzie k riziku;
4. Minimalizácia rizika bez ohľadu na očakávaný výnos;
5. Maximalizácia očakávaného výnosu bez ohľadu na riziko;
6. Minimalizácia očakávaného výnosu bez ohľadu na riziko.

Markowitz sa vo svojom modely zaoberal prvými troma bodmi tohto všeobecného problému. Historické ceny aktív pritom v jeho modely tvoria údajovú základňu, z ktorej sú vypočítané výnosy jednotlivých aktív. Výnosy založené na historických cenách aktív ďalej slúžia ako základ odhadu očakávaného výnosu portfólia a zároveň vstupujú do kovariančnej matice. Tá následne vystupuje ako súčasť minimalizačnej účelovej funkcie, ktorá bola navrhnutá Markowitzom a má tvar[3)]:

$$\min_{w \in R^n} \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} w_i w_j q_{ij} \tag{1}$$

$$\sum_{i=1}^{n} w_i \bar{r}^i = r_p \tag{2}$$

$$\sum_{i=1}^{n} w_i = 1 \tag{3}$$

Vektor $\bar{r}$ je vektor očakávaných výnosov aktív, w je vektor váh portfólia. Očakávaný výnos celého portfólia je teda súčin váh a očakávaných výnosov jednotlivých aktív. Ohraničujúca podmienka (2) poskytuje do rovnosti očakávaný výnos celého portfólia a požadovaný výnos r_p. Podmienka (3) zabezpečuje, že súčet váh všetkých zložiek portfólia bude rovný 1, q_{ij} v účelovej funkcii predstavuje kovarianciu medzi aktívom i a j v prípade, že $i \neq j$.

Takto stanovená úloha je však schopná poskytnúť optimálne riešenie iba v prípade, že sú splnené nasledovné predpoklady:

1. Prvky vektora $\bar{r}^i$ sú lineárne nezávislé;
2. Matica Q s prvkami q_{ij} je regulárna a kladne definitná;
3. Existujú dve akcie i a j, ktoré spĺňajú že: $\bar{r}^i \neq \bar{r}^j$, $i,j = 1,2, \ldots, n$.

Pokiaľ tieto navzájom súvisiace predpoklady nie sú splnené, nájdenie jednoznačného riešenie nie je možné a z $\bar{r}^i$ musí byť odstránené aktívum nespĺňajúce tieto predpoklady.

2) http://www.nag.co.uk/doc/techrep/pdf/tr2_00.pdf, platné ku dňu 12.2.2013

3) Melicherčík I., Olšarová L., Úradníček V.: Kapitoly z finančnej matematiky. Bratislava: EPOS.2005.

Aj keď Markowitzov model v priestore výnos- rozptyl sa stal základom pre ďalšie skúmanie divirzefikácie rizika v portfóliu, na základe svojich obmedzení, ktoré ho vzďaľujú od praktických potrieb hospodárskeho prostredia, sa jedná skôr o základný návrh na riešenie problému stanovenia skladby portfólia. Tento model bol ďalej rozširovaný a obohatený o rôzne svoje modifikácie, ako napríklad o zavedenie transakčných nákladov, nakoniec však ustúpil iným metódam výberu portfólia, ktorým sa budeme venovať neskôr.

2. Value at Risk

Value at Risk (VaR) predstavuje inú formu miery rizika, než pôvodne navrhoval Markowitz. VaR udáva hraničnú hodnotu, pre ktorú platí, že výnos bude na základe zvolenej pravdepodobnosti ? % horší, než je táto hodnota VaR. Value at Risk teda môžeme označiť i ako pravdepodobnostné ohraničenie, ktoré je používané pri výbere portfólia. Problematike VaR sa vo svojej práci venoval Charnes (1958). VaR sa stala štandardným nástrojom na riadenie rizika vo finančnej sfére a to hlavne kvôli jej koncepčnej a výpočtovej jednoduchosti[4].

Value at Risk vychádza z pohybov sadzieb aktív obchodovaných na trhu. Keďže sa riadi pravdepodobnostným rozdelením a určuje akú maximálnu pravdepodobnú stratu môže portfólio zaznamenať od dnes (čas *t*) po nasledujúcich *T* období. Na základe už spomínaného môžeme teda hovoriť o *1-α* percentnom VaR s časovým horizontom *T*. VaR je teda také číslo, pre ktoré plaí[5]:

$$P\left(h\left(f_t, \ldots, f_{t+T}, t+T\right) - h\left(f_t, t\right) < -VaR\right) = \alpha, \qquad (4)$$

kde *P* označuje pravdepodobnosť, f_{t+i} predstavuje náhodný vektor s rozmerom *n*, ktorý popisuje ceny, ktoré sú na trhu v čase *t+i*. *h* je funkcia ocenenia, ktorá stanovuje hodnotu portfólia na základe trhových cien aktív v konkrétnom čase a α je zvolená hladina pravdepodobnosti, pričom sa táto hladina zvykne používať z intervalu $\langle 1\%, 5\% \rangle$. Premenná *T* predstavuje časový horizont trvania investície. V klasickom VaR modely sa časť $h(f_t, \ldots, f_{t+T}, t + T)$ nahrádza pre účely zjednodušenia výpočtov ako $h(f_{t+T}, t)$. Nie je pritom nutné započítať náklady na držanie portfólia a ani uvažovať o zmenách cien na trhu, ktoré sa udiali počas obdobia *t+T*.

Pre skúmanie VaR sa zväčša postupuje v niekoľkých krokoch:

1. Voľba náhodného rozdelenia popisujúceho ceny aktív;
2. Odhad parametrov tohto rozdelenia na základe informácií poskytnutých trhom;
3. Simulácia realizácií náhodných vektorov;
4. Určenie VaR ako výberového kvantilu.

Takto popísaná metóda sa označuje ako simulácia Monte Carlo a predstavuje najuniverzálnejší prístup k výpočtu VaR[6].

Pri použití VaR sa pritom vychádza z predpokladu, že aktíva sa riadia normálnym rozdelením, poprípade sa vychádza z historických simulácií, ktorých výhodou je, že sa neriadia žiadnym konkrétnym rozdelením, ktorého parametre by bolo nutné odhadovať.

---

4) Pekár, J., Brezina, I., Čičková, Z.: Portfolio return, taking into account the costs of financial transactions. Quantitative methods in economics : multiple criteria decision making XVI. Bratislava: EKONÓM. 2012, pp. 171-175

5) Strnad, P.:Riziko tržní likvidity a jeho zohlednení v ukazovateli Value at Risk. Acta Oeconomica Pragensia 2/2009. Praha: VŠE. 2009

6) tamtiež

Napriek tomu, že VaR je v praxi veľmi používaná, podlieha táto metóda stále väčšej kritike odbornej verejnosti.

Kritika VaR

Medzi hlavných kritikov VaR patria Artzner (1999), Basak – Shapiro (2001). Napríklad Artzner (1997, 1999) uviedol tieto nedostatky:

1. VaR meria iba percentil ziskov a strát, a tak neberie do úvahy straty nad úroveň VaR,
2. VaR nie je koherentná miera rizika, lebo nie je sub-aditívna.

Títo autori zároveň uvádzajú, že VaR nie je vhodným prostriedkom na meranie rizika, nakoľko nezaručuje, že funkcia pozícií na trhu bude rýdzo konvexná a bude mať teda iba jeden extrém, ktorý by zaručoval nájdenie optimálneho riešenia.

Spomedzi českých autorov patrí medzi kritikov Value at Risk napríklad Strnad (2007), ktorý uvádza ako hlavné príčiny nevhodnosti použitia VaR nasledovné[7]:

1. VaR necharakterizuje veľmi málo pravdepodobné straty;
2. VaR nie je subaditívne;
3. VaR nehľadí dopredu;
4. VaR neuvažuje náklady likvidácie;
5. VaR je statické.

Aj keď niektoré z týchto nedostatkov sa už podarilo odstrániť, kritika, ktorej VaR podlieha zo strany najmä Artznera a Basaka – Shapira nebola vyvrátená a viedla k hľadaniu ďalších spôsobov zostavenia portfólia.

3. Capital Asset Pricing Model

Capital Asset Market Model (CAPM) ja model, ktorý bol navrhnutý Treynorom, Sharpem, Lintnerom a Mossinom ako iný prístup ne vyjadrenie vzťahu medzi rizikom a výnosom z finančných aktív. Základom tohto modelu je vytvorenie Security Market Line (SML), ktorá umožňuje vypočítanie miery výnos- riziko pre akékoľvek aktívum vo vzťahu k celému trhu. Dôležitým je pritom faktor β, ktorý nám udáva sklon SML. Miera výnos- riziko na trhu je v skutočnosti prémia za riziko, ktoré na trhu existuje. Zoskupením aktív nad úroveň rovnice sa rieši $E(R_i)$, pričom môžeme pozorovať už samotný CAPM (5) .

$$E(R_i) = R_f + \beta_i(E(R_m) - R_f), \tag{5}$$

kde:

$E(R_i)$ *– očakávaný výnos aktíva,*

R_f *– je bezriziková investícia,*

β_i – (v praxi taktiež zaužívaný jednoduchý názov Beta), citlivosť na zvýšenie očakávaného výnosu aktíva na očakávaný výnos na trhu, alebo taktiež (6),

$$\beta_i = \frac{Cov\left(R_i, R_m\right)}{Var\left(R_m\right)} \tag{6}$$

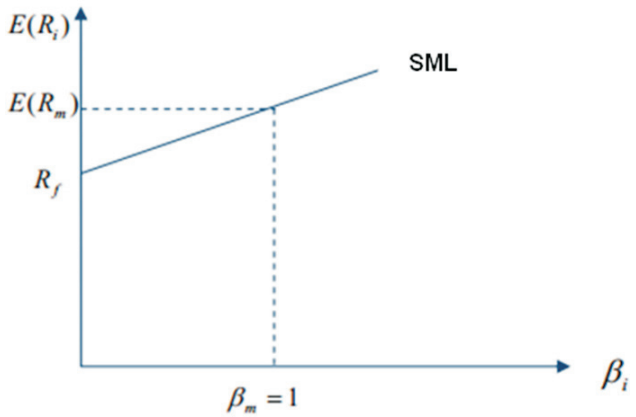
$E(R_m)$ *– očakávaný výnos na trhu,*

7) http://www.risk-management.cz/clanky/PetrStrnad-ValueAtRisk.pdf, platné k 23.1.2013

$E(R_m) - R_f$ – *prémia za trhové riziko,*

$E(R_i) - R_f$ – *prémia za jedinečné riziko.*

Pokiaľ upravíme tieto riziká, zistíme že platí vzťah (6), ktorý stanovuje, že jednotlivá riziková prémia sa rovná trhovej prémii krát β_i.



Obrázok 1: *Grafické zobrazenie SML s jej parametrami*

Model CAPM je zjednodušeným obrazom fungovania trhu a správania sa aktív na ňom. Práve pre tento fakt už od svojho publikovania podlieha kritike, ktorá mu vyčíta najmä:

1. Model predpokladá, že sa jedná o dokonalý trh;
2. predpokladá existenciu bezrizikových sadzieb na trhu;
3. zanedbáva dane a transakčné náklady, ktorým trh reálne podlieha a poplatky, ktoré sú neoddeliteľnou súčasťou súčasných trhov;
4. všetci investori majú rovnaké očakávania a averzia k riziku je u každého investora rovnaká.

Práve na základe tejto kritiky ani tento model nebol prijatý ako vhodné konečné riešenie problému výberu portfólia, no stal sa podstatným pri tvorbe ďalších prístupov zaoberajúcich sa touto problematikou.

4. Mean Absolute Deviation (MAD)

Mean Absolute Deviation je prístup, ktorý bol navrhnutý Konnom a Yamazakim (1991). Ako píšu autori: *„Modely rovnováhy majú zaviesť niekoľko nerealistických predpokladov na deriváciu základných vzťahov medzi mierou výnosnosti jednotlivých aktív a medzi trhovým portfóliom“*[8]. Toto tvrdenie je priamou kritikou CAPM a VaR.

Na základe svojej kritiky autori navrhli model, ktorý je postavený na základe nahradenia rozptylu absolútnou odchýlkou miery návratnosti aktív, pomocou ktorej je merané riziko portfólia. Výhodou tejto metódy je však najmä možnosť transformácie funkcie rizika na úlohu lineárneho programovania. Model ako taký sa stretol s pozitívnym ohlasom a stal sa jednou z v súčasnosti preskúmavaných metód výberu portfólia.

8) Konno, H., Yamazaki, H.: Mean-Absolute Deviation Portfolio Optimization Model and Its Applications to Tokyo Stock Market. Management Science, Vol. 37, No. 5, pp. 519-531, 1991

9) Zenios, S. A.: Practical financial optimization, Decision Mmaking for Financial Engineers. Padstow: Blackwell Publishing. 2007.

Ako už bolo spomenuté výhodou MAD je, že sa dá zapísať ako model lineárneho programovania v tvare[9]:

$$\min \sum_{l \in \Omega} p^l y^l \tag{7}$$

$$\sum_{i=1}^{n} \overline{P}_i x_i \geq r_p V_o \tag{8}$$

$$y^l \geq V(x; P^l) - V(x; \overline{P}), \quad \text{pre všetky } l \in \Omega \tag{9}$$

$$y^l \geq V(x; \overline{P}) - V(x; P^l), \quad \text{pre všetky } l \in \Omega \tag{10}$$

$$\sum_{i=1}^{n} P_{oi} x_i = V_o, \tag{11}$$

Model je zostavený tak, že podmienky (8) nám udávajú aká má byť priemerná budúca hodnota portfólia s cieľovým výnosom r_p. MAD ďalej obsahuje premenné y^l, ktoré sú odchýlkami od hodnôt portfólia a merajú horný a dolný potenciál odchýlenia sa od strednej hodnoty portfólia (9) a (10). Rozpočtové kritérium, ktoré má investor rovnako obmedzené v MAD vystupuje v (11) a zabezpečuje, aby úvodná investícia nebola väčšia než finančné možnosti investora.

Medzi výhody MAD patrí predovšetkým možnosť zápisu ako úlohy lineárneho programovania, čo umožňuje i riešenie úloh väčších rozmerov a znižuje tak potrebu zabezpečenie špeciálnou výpočtovou technikou[10]. Zároveň MAD ako úloha lineárneho programovania zvyšuje možnosť nájdenia optimálneho portfólia, ktoré bude pozostávať iba z jedného možného portfólia. Takto stanovený model umožňuje i väčšiu konkretizáciu problému, než modely, ktoré počítajú s dolnou hranicou rizika, pretože absolútna odchýlka môže byť považovaná za dobrú mieru dolnej hranice rizika[11].

I keď MAD modelu sa podarilo odstrániť hlavné nedostatky, ktoré boli kritizované Artznerom a Basakom – Shapirom pri VaR modely, tento základný model stále nezohľadňuje dynamický vývoj trhu.

5. Conditional Value at Risk (CVaR)

Ako už bolo spomenuté hlavnými kritikmi používania VaR ako miery na meranie rizika boli Artzner (1999) a Basak – Shapiro (2001). Títo autori však neostali iba pri kritike, ale pokúsili sa navrhnúť i nástroje, ktoré by boli schopné týmto nedostatkom VaR predísť. Na zmiernenie problémov, ktoré prináša VaR, je navrhnutá alternatívna miera rizika, ktorá sa nazýva Conditional Value at Risk (CVaR), prípadne niekde uvádzaná aj ako Average Value at Risk (AVaR). Keďže je to koherentná miera rizika, v úlohách výberu portfólia môže byť transformovaná na úlohu lineárneho programovania[12].

10) Konno, H., Koshizuka, T.: Mean-absolute deviation model. IIE Transactions Vol. 37, pp. 893-900.2005

11) Zenios, S. A.: Practical financial optimization, Decision Mmaking for Financial Engineers. Padstow: Blackwell Publishing. 2007.

12) Pekár, J., Brezina, I., Čičková, Z.: Portfolio return, taking into account the costs of financial transactions. Quantitative methods in economics : multiple criteria decision making XVI. Bratislava: EKONÓM. 2012, pp. 171—175

13) Pekár, J., Brezina, I., Čičková, Z.: Portfolio return, taking into account the costs of financial transactions. Quantitative methods in economics : multiple criteria decision making XVI. Bratislava: EKONÓM. 2012, pp. 171-175

Na zmiernenie problémov VaR, Artzner navrhol CVaR. CVaR je definovaná ako očakávaná strata presahujúca hodnotu VaR. Na základe tejto definície CVaR berie do úvahy hodnoty strát, ktoré sú za hodnotou VaR. Výhoda CVaR je v sub-aditivite, ktorá zaisťuje súdržnosť (koherenciu) ako mieru rizika. Na základe týchto poznatkov sa niektorý investori prikláňajú k CVaR[13].

VaR je mierou, ktorá môže investora zneistiť v jeho rozhodovaní a priviesť ho k zlým záverom, pokiaľ sa snaží maximalizovať svoj úžitok. Value at Risk by teda mohla spôsobiť, že ak by bola použitá ako jediná miera rizika nájdené portfólio, ktoré by sa javilo ako ideálne by v konečnom dôsledku spôsobilo stratu. CVaR dokáže predísť takým veľkým stratám, ako pri použití klasického VaR modelu, nakoľko berie i straty, ktoré sú nad úrovňou VaR. Ďalšou výhodou CVaR je, že umožňuje zapísanie úlohy výberu portfólia ako úlohu lineárneho programovania, čo zjednodušuje nájdenie riešenia. Efektívnosť techniky CVaR možno pripísať práve použitím techniky lineárneho programovania. Výhodou úloh lineárneho programovania je, že existujú mnohé programové balíčky, ktoré dokážu riešiť úlohy veľkých rozmerom s počtom až niekoľko tisíc premenných. Senzitivita parametrov je vypočítaná automaticky na základe premenných duálnej úlohy. Lineárne programovanie tiež umožňuje nájdenie celočíselného problému, čo pri úlohách výberu portfólia predstavuje veľkú výhodu, najmä pokiaľ sa jedná o financovanie na trhoch, kde je možné pracovať iba s balíkmi aktív. Umožňuje to taktiež zníženie náročnosti na výpočtovú techniku, ktorá je nevyhnutná pri riešení tohto typu.

Úloha CvaR je však riešiteľná i pokiaľ sa jedná o úlohu kvadratického programovania. Umožňujú to hlavne metódy vnútorných bodov, ktoré sú stále aplikovateľnejšie i na úlohy väčších rozmerov a to najmä vďaka rastúcej výkonnosti výpočtovej techniky[14]. Tieto techniky umožňujú i výpočty úlohy lineárneho programovania a tak sa považujú za všeobecnejší prístup k riešeniu CVaR.

Numerické experimenty naznačujú, že obvyklá minimalizácia CVaR vedie blízko k optimálnemu riešeniu, ktoré poskytuje VaR, nikdy však nepresiahne hodnotu VaR[15].

Cieľom úlohy je nájsť takú hodnotu optimálnych váh, pri ktorej hodnota rizika CVaR bude čo najnižšia. Teda CVaR je riešením nasledovnej optimalizačnej úlohy[16]:

$$CVaR_{\alpha}(X)=\min\left\{VaR_{\alpha}+\frac{1}{\alpha}E\left[\left(E_p-X-VaR_{\alpha}\right)^{+}\right]\right\}, \qquad (12)$$

kde VaR_{α} – Value at risk, E_p – cieľový výnos, výraz $(E_p - X - VaR)^+$ je kladná časť rozdielu $(E_p - X - VaR$, teda.

Pokiaľ uvažujeme o portfóliu, ktoré bude tvorené n aktívami a ich váhami $w = (w_1, w_2, \dots, w_n)$, ktoré nám určujú zloženie konečného portfólia, ďalej uvažujeme vektor výnosov $r_k = (r_{1,k}, r_{2,k}, \dots, r_{n,k})$, ktorý je zostavený pre k-ty scenár, tak budúci výnos portfólia môžeme určiť ako násobok vektora váh a vektora výnosov aktív $e^T r_k$, pričom očakávaný výnos predstavuje hodnotu $E[e^T r_k]$.

Vzťah (12) je teda možné vyjadriť ako:

$$CVaR_{\alpha}(w^T r)=\min\left\{VaR_{\alpha}+\frac{1}{\alpha}E\left[\left(E_p-w^T r_k-VaR_{\alpha}\right)^{+}\right]\right\}. \qquad (13)$$

14) Duarte, A.: Fast Computation of Efficient Portfolios, Journal of Risk 1(4), pp.71-94, 1999

15) Rockafellar, R.T., Uryasev, S.: Optimization of Conditional Value-at-Risk. The Journal of Risk. Vol. 2, No. 3, pp. 21-41, 2000

16) Pekár, J., Brezina, I., Čičková, Z.: Portfolio return, taking into account the costs of financial transactions. Quantitative methods in economics : multiple criteria decision making XVI. Bratislava: EKONÓM. 2012, pp. 171-175

Pre hodnoty týchto scenárov stanovíme predpoklad, že nerastú s rovnakou pravdepodobnosťou. Pokiaľ *t* predstavuje počet riadkov matice scenárov, tak účelová funkcia zostavená na základe tohto predpokladu bude vyzerať nasledovne:

$$CVaR_{\alpha}(X)=\min\left\{VaR_{\alpha}+\frac{1}{\alpha}E\left[\left(E_p-X-VaR_{\alpha}\right)^{+}\right]\right\}, \tag{14}$$

V takto zostavenej úlohe však stále vzniká možnosť nelineárnosti úlohy. Aby sme tomu zabránili, je potrebné nehradiť člen $[E_p - w^T r_k - VaR_\alpha]^+$ premnnou $z = (z_1, z_2, \dots, z_t)$, ktorá bude nezáporná. Potom možno konečnú úlohu lineárneho programovania s novou premennou $z_k \geq [E_p - w^T r_k - VaR_\alpha]^+$ vyjadriť ako[17]:

$$\min\left\{VaR_{\alpha}+\frac{1}{\alpha t}\sum_{k=1}^{t}z_k\right\} \tag{15}$$

$$z_k - E_p + w^T r_k + VaR_\alpha \geq 0, \quad k = \{1,2, \dots, t\} \tag{16}$$

$$w^T E(r_n) \geq E_p, \tag{17}$$

$$w^T e = 1, \tag{18}$$

$$z \geq 0, \tag{19}$$

kde $E(r_n)$ je vektor očakávaných výnosov aktív.

Záver

Z metód, ktoré sú v tomto článku popísané vyplýva, že aj keď problematika výberu portfólia je oblasťou, ktorá púta pozornosť odborníkov už niekoľko desiatok rokov, nepodarilo sa spoľahlivo nájsť metódu, ktorá by zaručila nájdenie zaručene najlepšieho zloženia portfólia. Aj keď klasická metóda priemer- rozptyl a CAPM vytvorili matematický aparát, ktorým bolo možné riešiť problém výberu portfólia, ich následná kritika spochybnila ich vierohodnosť. Napriek tomu tieto spôsoby merania rizika patria v praxi k najobľúbenejším, najmä na základe svojej jednoduchosti. Metódy MAD a CVaR môžeme stále považovať za relatívne mladé prístupy k výberu portfólia. Ani tieto metódy však nie sú overené a nezohľadňujú určité aspekty trhu a preto sú stále predmetom skúmania a svojho rozširovania, najmä a parameter nákladov. Autor sa dnes venujú aj vytvoreniu dynamických modelov, ktoré vychádzajú práve z v tomto článku spomínaných modelov.

Použitá literatúra:

- [1] ARTZNER, P., DELBAEN, F., EBER, J. M., HEATH, D.: "Thinking Coherently," Risk, 10 (11), pp. 68–71, 1997
- [2] ARTZNER, P., DELBAEN, F., EBER, J., and HEATH, D.: Coherent measure of risk. Mathematical Finance, 9 (3):203–228, 1999
- [3] BASAK, S., SHAPIRO, A.: Value-at-risk-based risk management: optimal policies and asset prices, Rev. Financ. Stud. 14(2), 371-405, 2001
- [4] BREZINA, I. Jr., MIŠOVIČ, A.: Vplyv výberu miery rizika na zloženie portfólia, Mladá veda AIESA 2012 – Participácia doktorandov a mladých vedeckých pracovníkov na budovaní spoločnosti založenej na vedomostiach (zborník príspevkov), Bratislava: EKONÓM. 2012
- [5] DUARTE, A.: Fast Computation of Efficient Portfolios, Journal of Risk 1(4), pp.71-94, 1999

17) tamtiež

- [6] CHARNES, A., COOPER, W. W., SYMONDS, G. H.: Cost horizonts and certainty equivalents: an approach to stochastic programming of heating oil. Management Science 4, pp 235-263, 1958
- [7] KONNO, H., KOSHIZUKA, T.: Mean-absolute deviation model. IIE Transactions Vol. 37, pp. 893-900.2005
- [8] KONNO, H., YAMAZAKI, H.: Mean-Absolute Deviation Portfolio Optimization Model and Its Applications to Tokyo Stock Market. Management Science, Vol. 37, No. 5, pp. 519-531, 1991
- [9] MARKOWITZ, H.: The Journal of Finance, March 1952, pp. 77-91
- [10] MARKOWITZ, H.: Portfolio Selection -Efficient Diversification of Investment. New York, 1959.
- [11] MELICHERČÍK, I., OLŠAROVÁ, L., ÚRADNÍČEK, V.: Kapitoly z finančnej matematiky. Bratislava: EPOS.2005.
- [12] PEKÁR, J.: Konštrukcia množiny investičných príležitostí z fondov ESPA. Medzinárodný vedecký seminár mladých vedeckých pracovníkov Katedry ekonometrie FIS VŠE v Praze a Katedry operačného výskumu a ekonometrie FHI EU v Bratislave, Praha: 2007
- [13] PEKÁR, J.: Výber investičného fondu ESPA metódou Promethee. Využitie kvantitatívnych metód vo vedecko-výskumnej činnosti a v praxi VIII. Bratislava : EKONÓM, 2007
- [14] PEKÁR, J., BREZINA, I., ČIČKOVÁ, Z.: Portfolio return, taking into account the costs of financial transactions. Quantitative methods in economics : multiple criteria decision making XVI. Bratislava: EKONÓM. 2012, pp. 171-175
- [15] PEKÁR, J., ČIČKOVÁ, Z.: Expected equity returns in Slovakia using the Black- Litterman model. International journal of strategic management and decision support systems in strategic management. Vol. 16, No. 2, pp. 69-73, 2011
- [16] PEKÁR, J., PINDA, Ľ.: The portfolio optimization models for insurance. Quantitative methods in economics : multiple criteria decision making XIV. Bratislava: IURA EDITION, 2008
- [17] ROCKAFELLAR, R.T., URYASEV, S.: Optimization of Conditional Value-at-Risk. The Journal of Risk. Vol. 2, No. 3, pp. 21-41, 2000
- [18] STRNAD, P.: Riziko tržní likvidity a jeho zohlednění v ukazovateli Value at Risk. Acta Oeconomica Pragensia 2/2009. Praha: VŠE. 2009
- [19] ZENIOS, S. A.: Practical financial optimization, Decision Mmaking for Financial Engineers. Padstow: Blackwell Publishing. 2007.

Internetové zdroje:

- [1] http://www.nag.co.uk/doc/techrep/pdf/tr2_00.pdf, dostupné 12.1.2013
- [2] http://www.risk-management.cz/clanky/PetrStrnad-ValueAtRisk.pdf, dostupné 23.1.2013

Ing. Ivan Brezina Jr.,
Katedra operačného výskumu a akonometrie, Fakulta hospodárskej informatiky,
Ekonomická univerzita v Bratislave, Dolnozemská cesta 1/b, 852 35 Bratislava,
brezina.ivan@yahoo.com,
+421 2 67 295 826.

Využitie algoritmu včelej kolónie pri riešení optimalizačných úloh

Application of bee colony algorithm in solving optimization problems

Marián Šedivý

Abstract:

This paper aims on solving optimization problems, specially solving the vehicle routing problem for Bratislava Self-Governing Region, by use of bee colony algorithm. Bees are highly organized social insects. The survival of the entire colony depends on every individual bee. Bees use systematic task segregation among them to ensure a continues existence of their colony. By simulating behavior of bees during foraging we can efficiently find optimal solutions to optimization problems.

Keywords:

BCO, VRP, Bee Colony Optimization, Vehicle Routing Problem, Swarm Intelligence

ACM classification: G1.6, I.2.8, J.4

Úvod

Rozvozný problém (Vehicle Routing Problem – VRP), alebo aj problém okružných jázd radíme medzi najviac študované problémy kombinatorickej optimalizácie. Riešením tohto problému je nájdenie optimálneho diagramu ciest, po ktorom pôjdu jednotlivé vozidlá tak, aby uspokojili požiadavky všetkých zákazníkov s minimálnymi nákladmi na prepravu. Taktiež nesmieme zabudnúť na dodržanie všetkých obmedzujúcich podmienok, ktoré súvisia s povahou zadania tohto problému. Je potrebné uvažovať o počte centrálnych skladov, kapacite vozidiel, dopyte jednotlivých zákazníkov, ale aj o tom, či riešením úlohy je minimalizácia počtu vozidiel alebo celkovej vzdialenosti prejdenej všetkými vozidlami. Od roku 1959, kedy tento problém ako prví popísali vo svojej práci Dantzig a Ramser (Dantzig, et al., 1959), bolo venovaných veľa štúdií exaktným a aproximatívnym riešeniam tohto problému, ako aj jeho variácií.

Algoritmus včelej kolónie je inšpirovaný správaním včiel pri obstarávaní potravy. Toto správanie bolo aplikované ako výpočtový algoritmus na riešenie komplexných problémov v rôznych oblastiach. V literatúre sa spomína aplikácia algoritmu pri dynamickej alokácii serverov (Nakrani, et al., 2004), optimalizácii numerických funkcií (Karaboga, et al., 2007), (Karaboga, et al., 2008), smerovaní paketov[1)] v telekomunikačných sieťach (Wedde, et al., 2004), stochastickom

1) Paket je ucelený blok dát, ktorý sa prenáša v sieti a spracováva ako celok. Je nedeliteľný na menšie časti.

VRP (Lučić, et al., 2003), (Teodorović, 2008) a rozvrhovacom probléme – Job Shop Scheduling Problem(Chong, et al., 2006), (Chong, et al., 2007).

Súčasný stav problematiky riešenia okružných úloh

VRP pokrýva distribúciu tovarov alebo služieb v danom časovom horizonte pre skupinu zákazníkov, pomocou súboru vozidiel, ktoré sú umiestnené v jednom alebo viacerých centrálnych skladoch, obsluhované posádkou a jazdiace po vhodnej cestnej sieti. Konkrétne riešenie VRP spočíva v nájdení množiny trás, z ktorej každá prislúcha jednému vozidlu tak, aby začalo a skončilo svoju cestu v prislúchajúcej stanici, boli uspokojené všetky požiadavky zákazníkov a splnené všetky obmedzujúce podmienky pri minimalizácii globálnych cestovných nákladov.

Zložitosť VRP problému vo veľkej miere vyplýva z faktu, že VRP problém sa nachádza na rozhraní dvoch dobre známych a študovaných problémov: problému batohu (Bin Packing Problem – BPP[2)]) a problému obchodného cestujúceho (Travelling Salesman Problem – TSP[3)]). Pre vyriešenie VRP potrebujeme vyriešiť obidva tieto problémy. Treba podotknúť, že tieto problémy patria do triedy NP – ťažkých problémov a nájdenie riešenie pre každý z nich osobitne vyžaduje veľkú námahu a je časovo náročné.

Sieť ciest využívaná na transport tovarov a služieb je vo všeobecnosti opísaná ako graf, kde jeho uzly predstavujú stanice, zákazníkov alebo križovatky a hrany (spojnice týchto uzlov) sú úseky trás. Tieto úseky, ako aj celé grafy môžu byť orientované alebo neorientované. Orientované grafy sú typické pre mestské trasy, kde je potrebné rešpektovať jednosmerné ulice alebo obchádzky. Každá hrana má svoje ohodnotenie vyjadrené napríklad ako dĺžku v km, resp. ako čas potrebný na jej prejdenie (ten závisí od použitého vozidla, ale aj od času, kedy daným úsekom prechádzame).

Schéma klasifikácie variantov problému okružných jázd podľa (Genderau, et al., 1998)

Tabuľka 1: *Varianty problému okružných jázd*

Väzba / Kapacita	M:M[4)]	1:M[5)]
Obmedzená kapacita	Odvoz na telefón	Systém doplňovania
Neobmedzená kapacita	Expresné doručenie pošty	Služby kuriéra a opráv

Problémy typu M:M s kapacitnými obmedzeniami patria medzi najviac komplikované z toho dôvodu, že miesta vyzdvihnutia a doručenia musia ležať na jednej trase a vyzdvihnutie musí predchádzať doručeniu.

Pri probléme odvozu na telefón (Dial-a-Ride) taktiež známeho aj ako Stacker Crane Problem uvažujeme o dvoch variáciách. Pri preemptívnom type problému je možné nechať prepravované objekty na medzizastávke a neskôr ich opäť naložiť a doručiť, čím sa zefektívni celý proces. Pri nepreemptívnom type túto možnosť nemáme. Odvoz na telefón je často využívaný pri prevoze starších ľudí, prípadne ľudí s obmedzenou pohyblivosťou či zdieľaných taxíkoch.

2) Bin Packing Problem je problém, pri ktorom treba uložiť čo najviac objektov do čo najmenšieho počtu kontajnerov s konštantnou veľkosťou.

3) Travelling Salesman Problem – riešením problému obchodného cestujúceho je nájsť takú okružnú cestu medzi mestami, ktorá bude najlacnejšia. Okružná cesta značí poradie, v akom je potrebné prejsť všetkými miestami práve raz a na záver sa vrátiť do začiatočného miesta.

4) Každá požiadavka zahŕňa obe lokality (miesto vyzdvihnutia aj doručenia)

5) Každá požiadavka zahŕňa iba jednu lokalitu (doručenie alebo vyzdvihnutie)

Systém doplňovania môžeme charakterizovať ako lokálny odvoz na telefón, ktorý „doplňuje“ iný väčší transportný systém v určitej lokalite. Napríklad môže ísť o zozbieranie zákazníkov menším autobusom, ktorý ich následne odvezie k vlaku alebo na letisko.

Pri expresnom doručení pošty je dôležitým faktorom nutnosť aby vyzdvihnutie a doručenie pošty v danej oblasti boli vykonané tým istým vozidlom.

Služby kuriéra a opráv môžeme prirovnať k problému okružných jázd s časovými oknami, kde vykonanie služby nemusí vždy nasledovať za požiadavkou, ale je centrálne naplánované tak, aby sa minimalizovali náklady na prepravu. Taktiež veľmi dôležitú úlohu zohráva čas potrebný na vykonanie služby.

Algoritmus včelej kolónie

Tento algoritmus, často označovaný aj ako optimalizácia prostredníctvom včelej kolónie (Bee Colony Optimization – BCO), alebo umelá včelia kolónia (Artificial Bee Colony – ABC), simuluje správanie včiel pri zbere potravy. Keďže lietanie je pre včely veľmi náročné a vyčerpávajúce, snažia sa optimalizovať zber potravy, aby získali čo najväčšie množstvo peľu preletením najkratšej vzdialenosti. Základnou ideou BCO je vytvorenie multi – agentového systému (kolónie umelých včiel), ktorý je schopný riešiť zložité problémy kombinatorickej optimalizácie.

Správanie včiel pri obstarávaní potravy bolo dlhé roky záhadou, až kým Karl von Frisch[6)] nedešifroval význam pohybov, ktoré robia včely pri svojom kývavom tanci (waggle dance). Tento tanec včely používajú na vzájomnú komunikáciu. Predpokladajme, že včela našla bohatý zdroj potravy. Po jej návrate do úľa začne tancovať v tvare osmičky. Prostredníctvom tohto informatívneho tanca včela dokáže informovať ostané zberačky potravy o smere a vzdialenosti k novoobjavenému zdroju a tým k nemu môže pritiahnuť viac včiel.

Celý algoritmus, založený na procese obstarávania a zberu potravy, môžeme opísať pomocou nasledovných krokov[7)]:

- Počas prvej iterácie, keď ešte včely nemôžu sledovať žiaden tanec, vyletia z úľa náhodne do miest s potravou. Vo svojej práci (Wong, et al., 2009) využívajú pre túto fázu dva heuristické prístupy a to heuristiku prechodu stavov[8)] a heuristiku najbližšieho suseda[9)],
- Následne sa včely vrátia do úľa, kde odovzdajú potravu a majú na výber z viacerých možností:
  - zavrhnúť daný zdroj potravy a stať sa nerozhodnutým nasledovníkom,
  - pokračovať v zbere potravy na danom zdroji bez toho, aby presvedčili ďalšie včely,
  - začať tancovať a tým presvedčiť ostané včely nasledovníčky, aby išli s nimi zbierať potravu na danom zdroji,

Včela si vyberá z uvedených možností na základe určitej pravdepodobnosti. Mechanizmus, kedy si včela vyberá nasledovanie určitej tancujúcej včely, nie je veľmi známy, ale existujú domnienky, že regrutovanie včiel je funkciou kvality daného zdroja (Camazine, et al., 1991).

---

6) Podrobný opis experimentov, ktoré vykonal Karl Von Frisch pre dešifrovanie správania včiel je dostupný na http://www.nobelprize.org/nobel_prizes/medicine/laureates/1973/frisch-lecture.pdf

7) Daný postup bol odvodený od formulácií použitých v prácach autorov (Wong, et al., 2009) a (Karaboga, et al., 2008).

8) Včela začína v náhodnom mieste a na základe pravidla prechodu stavov si vyberie ďalšie miesto kam poletí.

9) Včela začína v ľubovoľnom mieste a vyberá si najbližšie susedné miesto kam poletí. Ak tejto podmienke vyhovuje viac miest, aplikuje sa náhodný výber.

Konštrukcia cesty umelými včelami

Počas fázy zberu potravy musí včela prejsť z jedného miesta do druhého miesta pokiaľ nevykoná celý okruh (vráti sa na začiatok svojej cesty) na základe tranzitívneho pravidla. Toto pravidlo definuje pravdepodobnosť prechodu z miesta i do miesta j po n prechodoch, p_{ij},n ako je zapísané vo vzťahu (1.1). Keď je konštrukcia cesty hotová, získame permutáciu miest, ktorá je jedným z riešení nášho VRP. Označme d_{ij} vzdialenosť dvoch miest a ρ_{ij} , n fitness hodnotu hrany spájajúcu tieto dve mestá. Nech $A_{i,n}$ je množina zatiaľ nenavštívených miest, ktoré sú dosiahnuteľné z miesta i na prechode n. Parametre α a β určujú relatívnu významnosť fitness hodnoty hrany verzus heuristickú vzdialenosť.

$$P_{ij,n} = \frac{\left[\rho_{ij,n}\right]^{\alpha} \cdot \left[\frac{1}{d_{ij}}\right]^{\beta}}{\sum_{j \in A_{i,n}} \left(\left[\rho_{ij,n}\right]^{\alpha} \cdot \left[\frac{1}{d_{ij}}\right]^{\beta} \right)} \qquad (1.1)$$

Vzdialenosť d_{ij} nadobúda hodnoty nepriamo úmerné k $P_{ij,n}$. Čím je vzdialenosť medzi i a j kratšia, tým väčšia je pravdepodobnosť, že j bude vybrané ako nasledujúce miesto návštevy. Fitness hodnota hrany $P_{ij,n}$ meria pravdepodobnosť, že v preferovanej ceste bude existovať spojenie medzi i a j. V prírode ak včela nájde nový zdroj potravy, začne tancovať. Ak týmto tancom osloví inú včelu, tá bude zbierať potravu na novom zdroji. Pri BCO je táto situácia modelovaná implementáciou preferovanej cesty, ktorú označíme θ. θ je permutáciou prechodov jednotlivými miestami, ktoré včela získala od inej včely (navádzanie pri obstarávaní potravy). Ďalej uvedieme vzťah pre $P_{ij,n}$:

$$\rho_{ij,n} = \begin{cases} \lambda & , \; j \in F_{i,n} \,, \left|A_{i,n}\right| > 1 \\ \dfrac{1-\lambda\left|A_{i,n} \cap F_{i,n}\right|}{\left|A_{i,n} \cap F_{i,n}\right|} & , \; j \notin F_{i,n} \,, \left|A_{i,n}\right| > 1 \\ 1 & , \; \left|A_{i,n}\right| = 1 \end{cases} \qquad (1.2)$$

$$\forall j \in A_{i,n} \,, \qquad 0 \le \lambda \le 1$$

kde λ reprezentuje pravdepodobnosť navštívenia miesta vo θ. $F_{i,n}$ je množina, ktorá obsahuje jedno preferované miesto odporučené v θ, do ktorého sa včela presunie z miesta i na prechode n. Ak včela v nultom kroku začala prehľadávanie z úľa H, potom položíme $F_{H,0} = \{\theta(1)\}$. Ak práve navštívené miesto i je na m-tej pozícii v preferovanej ceste po n prechodoch, potom $F_{\theta(m),n} = \{\theta(m+1)\}$. $F_{i,n}$ obsahuje iba jedno miesto, pretože uvažujeme iba o $\theta(m+1)$ (nasledujúce priľahlé miesto k miestu i z θ). Prvé dve podmienky z (1.2) zabezpečujú, že hrany odporúčané v θ sú priradené s pravdepodobnosťou λ, zatiaľ čo zvyšné hrany sú priradené s rovnakou pravdepodobnosťou. Ak uvažujeme o l hranách, jedna hrana (ktorá existuje v θ) sa bude rovnať λ a ostatné $\frac{(1-\lambda)}{l-1}$. Ak žiadna hrana nie je podobná v porovnaní s θ, všetky uvažované hrany z $A_{i,n}$ sa budú rovnať $\frac{\cdot}{l}$. Tretia podmienka priradí hodnotu 1 pre $\rho_{ij,n}$ ak už ostáva iba jedno miesto z $A_{i,n}$. Ide teda o posledný prechod predtým, než sa včela vráti do začiatočného miesta a tým ukončí svoju trasu.

Opis práce algoritmu

Včely v úle sú rozdelené do troch skupín a to aktívne včely, neaktívne včely a prieskumníci. Aktívne včely sú všetky, ktoré práve využívajú zdroj potravy. Neaktívne včely sú v úli a čakajú na

informácie, od aktívnych včiel. Na základe kvality nájdeného zdroja jedla, aktívne včely tancujú pre neaktívne, ktoré si potom vďaka týmto informáciám vyberajú zdroj jedla kam poletia. Lepšie zdroje preto pritiahnu viac včiel. Prieskumníci sú včely, ktoré hľadajú nové zdroje potravy v okolí úľu.

- **Aktívna včela (Foraging)**: generuje susedné riešenia k svojmu aktuálnemu riešeniu, ktoré má práve v pamäti,
- **Neaktívna včela (Observing)**: čaká v úli a na základe tancovania aktívnych včiel a prieskumníkov, môže s určitou pravdepodobnosťou prijať niektoré z ich riešení. Aktívna včela, ktorá pre zadaný počet iterácií nedokázala zlepšiť svoje riešenie sa stáva neaktívnou a zároveň sa náhodná neaktívna včela aktivuje.
- **Prieskumník (Scouting)**: generuje náhodné riešenia, a tým je zabezpečené vyviaznutie z lokálneho extrému.

Každá včela má vo svojej pamäti riešenie, ktoré reprezentuje jednotlivé trasy pre všetky vozidlá. Ide teda vždy o kompletné riešenie. Aktívne včely používajú na nájdenie susedného riešenia metódu GetNextSolution(), ktorá sa skladá z dvoch krokov SwapCitiesInOneRoute() a SwapCitiesAmongRoutes(). Prieskumníci generujú náhodné riešenia úplne nezávislé na ich aktuálnom. Ak nájde aktívna včela alebo prieskumník lepšie riešenie ako mala doteraz, aktualizuje svoje riešenie, zavolá metódu DoWaggleDance() a zároveň ho porovná s globálnym najlepším riešením. Ak je nájdené riešenie lepšie ako globálne, tak ho aktualizuje. Prácu jednotlivých včiel a pomocné metódy môžeme vyjadriť nasledovne:

Aktívna včela

Aktívna včela sa snaží prehľadať okolie najbližšie k svojmu aktuálnemu riešeniu, čo je zabezpečené metódou GetNextSolution(). Taktiež je tu zavedená pravdepodobnosť chyby ProbabilityMistake, keď včela môže odmietnuť lepšie riešenie, alebo aj prijať horšie. Ide o veľmi malé číslo, napríklad 0,01, čiže včela sa pomýli s pravdepodobnosťou 1 percento. Ak aktívna včela našla lepšie riešenie ako mala doteraz, zavolá metódu DoWaggleDance(), pomocou ktorej sa snaží presvedčiť neaktívne včely, aby ju nasledovali. Pokiaľ včela nedokáže zlepšiť svoje riešenie počas zadaného počtu návštev jej zdroja MaxVisits, stane sa z nej neaktívna včela a zároveň sa náhodná neaktívna včela aktivuje. Niektorí autori uvádzajú, že maximálny počet návštev by mal byť rovný približne päť násobku počtu miest.

Prieskumník

Na rozdiel od aktívnej včely, prieskumník nerobí chyby. To znamená, že prijme riešenie len vtedy, ak je naozaj lepšie ako to, ktoré mal doteraz v pamäti. Následne zavolá metódu DoWaggleDance(). Pomocou včiel prieskumníkov je zabezpečené generovanie náhodných riešení a tým je algoritmu umožnené vyviaznutie z lokálneho extrému.

Neaktívna včela

Neaktívna včela v našom prípade nerobí nič, ako už názov napovedá. Bolo by možné implementovať napríklad metódu GetNextSolution() aj pre neaktívnu včelu a získať tak susedné riešenie k jej aktuálnemu riešeniu.

Tancovanie

Metóda DoWaggleDance() umožňuje s určitou pravdepodobnosťou ProbabilityPersuation prijať neaktívnej včele riešenie od práve tancujúcej včely. Napríklad ak by bola použitá pravdepodobnosť 0,9, znamenalo by to, že s pravdepodobnosťou 90 percent bude neaktívna včela nasledovať práve tancujúcu včelu. Túto metódu využívajú na propagovanie dobrých riešení aktívne včely a prieskumníci a je ňou zabezpečené odovzdávanie informácií medzi včelami v úle.

Generovanie susedných riešení

Metóda GetNextSolution() využíva ďalšie dve metódy pre vygenerovanie susedného riešenia. Zároveň je táto metóda vložená v cykle, ktorý kontroluje či vygenerované riešenie spĺňa obmedzujúce podmienky. Metóda je volaná dovtedy, kým vygenerované riešenie porušuje obmedzujúce podmienky. Tým je zabezpečené, že včely pracujú len s vyhovujúcimi riešeniami.

V prvej metóde SwapCitiesInOneRoute() sa v jednej trase premiestnia navzájom dve susedné miesta[10]. Napríklad pre postupnosť miest A,B,C,D,E,A je susedným riešením A,B,C,E,D,A. Ide vlastne o permutáciu pôvodného riešenia, avšak vzniká tu priestor pre polemizovanie, či susedné riešenie vzniká len výmenou dvoch priľahlých miest, alebo môžu byť vymenené dve ľubovoľné miesta napr. A,D,C,B,E,A.

V druhej metóde SwapCitiesAmongRoutes() vyberieme náhodne dve trasy z riešenia, ktoré má včela v pamäti. Z prvej trasy potom náhodne vyberieme jedno miesto a to vložíme na náhodnú pozíciu v druhej trase.

Generovanie náhodných riešení

Metóda CreateRandomSolution()generuje náhodné riešenie pre prieskumníkov. Jej algoritmus je nasledovný. Pre každé miesto vyber náhodnú trasu, ak sa dá miesto vložiť bez toho, aby porušilo obmedzujúcu podmienku (napríklad presiahlo kapacitu vozidla), tak vlož miesto do vybranej trasy. Ak sa miesto nedá vložiť, zvýš počítadlo neúspechu[11]. Pokiaľ sa počet neúspešných vložení rovná počtu vozidiel, tak poruš obmedzujúcu podmienku a vlož miesto na trasu. Metóda CreateRandomSolution() je volaná v cykle, ktorý kontroluje správnosť vygenerovaného riešenia dovtedy, kým porušuje obmedzujúcu podmienku.

Dosiahnuté výsledky

Opísaný algoritmus bol aplikovaný na riešenie problému okružných úloh pre Bratislavský samosprávny kraj. Problém bol špecifikovaný nasledovne:

- počet vozidiel – 5,
- kapacita vozidla – 40 000 jednotiek,
- dopyt jednotlivých miest – priamoúmerne odvodený od počtu obyvateľov,
- tesnosť – 88,84 %.
- umiestnenie centrálneho skladu v Čunove

V takomto prípade sa celková vzdialenosť prejdená všetkými vozidlami rovnala 652 km. Usporiadanie jednotlivých obcí medzi vozidlá zachytáva nasledujúca tabuľka Tabuľka 1 a graficky znázorňuje obrázok Obrázok 1.

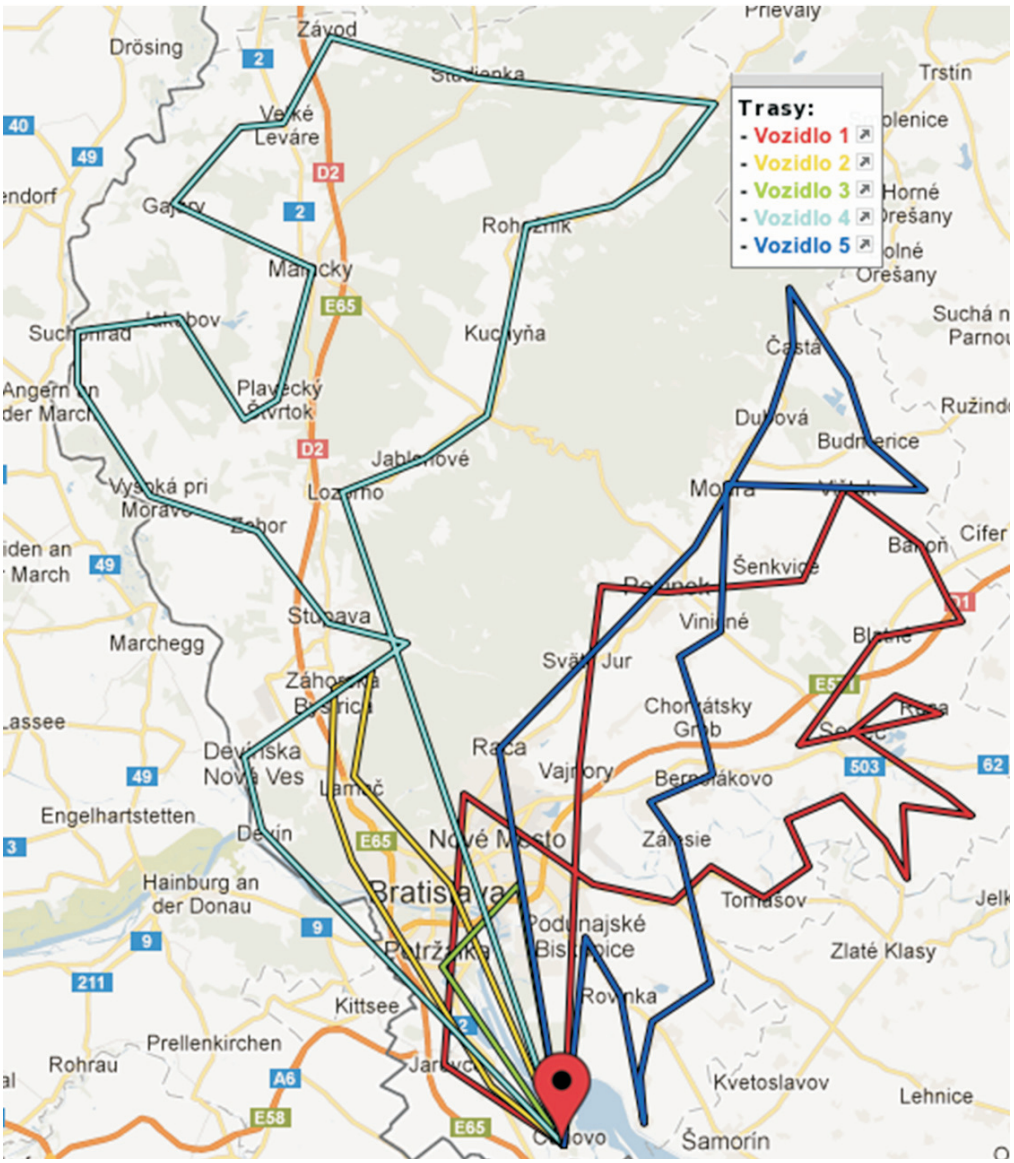
Tabuľka 1 *Rozdelenie obcí medzi vozidlá*

1. vozidlo	Bratislava – Čunovo, Bratislava – Jarovce, Bratislava – Nové Mesto, Bratislava – Vrakuňa, Most pri Bratislave, Malinovo, Tomášov, Vlky, Nová Dedinka, Tureň, Hrubý Šúr, Hurbanova Ves, Kostolná pri Dunaji, Hrubá Borša, Kráľová pri Senci, Senec, Boldog, Reca, Veľký Biel, Blatné, Čataj, Igram, Kaplna, Báhoň, Vištuk, Šenkvice, Pezinok, Limbach, Svätý Jur, Bratislava – Vajnory, Bratislava – Čunovo

10) Definícia existencie susedných miest je kľúčovou podmienkou pre riešenie úlohy VRP. Ak by totiž neexistoval predpoklad susednosti jednotlivých miest, včelí algoritmus by nebol schopný úlohu riešiť.

11) Počítadlo neúspechu zaznamenáva počet pokusov, kedy sa nepodarilo miesto vložiť na vybranú náhodnú trasu, pretože porušilo kapacitné obmedzenie vozidla.

2. vozidlo	Bratislava – Čunovo, Bratislava – Rusovce, Bratislava – Karlova Ves, Bratislava – Dúbravka, Bratislava – Záhorská Bystrica, Marianka, Bratislava – Lamač, Bratislava – Staré Mesto, Bratislava – Čunovo
3. vozidlo	Bratislava – Čunovo, Bratislava – Petržalka, Bratislava – Ružinov, Bratislava – Čunovo
4. vozidlo	Bratislava – Čunovo, Lozorno, Jablonové, Pernek, Kuchyňa, Rohožník, Sološnica, Plavecké Podhradie, Plavecký Mikuláš, Studienka, Závod, Veľké Leváre, Malé Leváre, Gajary, Kostolište, Malacky, Plavecký Štvrtok, Láb, Jakubov, Suchohrad, Záhorská Ves, Vysoká pri Morave, Zohor, Stupava, Borinka, Bratislava – Devínska Nová Ves, Bratislava – Devín, Bratislava – Čunovo
5. vozidlo	Bratislava – Čunovo, Bratislava – Podunajské Biskupice, Rovinka, Hamuliakovo, Kalinkovo, Dunajská Lužná, Miloslavov, Zálesie, Ivanka pri Dunaji, Bernolákovo, Chorvátsky Grob, Slovenský Grob, Viničné, Modra, Jablonec, Budmerice, Štefanová, Doľany, Píla, Častá, Dubová, Vinosady, Bratislava – Rača, Bratislava – Čunovo



Obrázok 1 *Trasy jednotlivých vozidiel*

Záver

Keďže riešenie problému okružných jázd by aplikovaním exaktných metód trvalo neúmerne dlho, aplikácia včelieho algoritmu sa javí ako veľmi vhodná. Algoritmus včelej kolónie je schopný efektívne riešiť optimalizačné úlohy a v prípustnom čase ponúknuť riešenia, ktoré nie sú veľmi vzdialené od optima.

Použitá literatúra:

- [1] CAMAZINE, S. and SNEYD, J. 1991. *A model of collective nectar source selection by honey bees: Self-organization through simple rules* . U.S.A. : Elsevier, 1991.
- [2] DANTZIG, G. B. and RAMSER, J. H. 1959. *The truck dispatching problem.* Filadelfia : Management Science, 1959.
- [3] GENDERAU, M. and POTVIN, J-Y. 1998. Dynamic Vehicle Routing and Dispatching. [book auth.] T. G. Crainic and G. Laporte. *Fleet management and logistics.* Boston : Kluwer, 1998.
- [4] CHONG, C. S., et al. 2006. *A bee colony optimization algorithm to job shop scheduling.* Arizona : Winter Simulation Conference, 2006 – 2007.*Using a bee colony algorithm for neighborhood search in job shop scheduling problems.* Praha : 21. European Conference on Modeling and Simulation, 2007.
- [5] KARABOGA, D. and BASTURK, B. 2008. *On the performance of artificial bee colony (abc) algorithm.* Turecko : Springer, 2008.
- [6] KARABOGA, D. and BSTURK, B. 2007. *A powerful and efficient algorithm for numerical function optimization: artificial bee colony (abc) algorithm.* Turecko : Springer, 2007.
- [7] LUČIĆ, P. and TEODOROVIĆ, D. 2003.*Vehicle routing problem with uncertain demand at nodes: The bee system and fuzzy logic approach.* Berlín : Springer-Verlag, 2003.
- [8] NAKRANI, S. and TOVEY, C. 2004. *On honey bees and dynamic server allocation in internet hosting centers.* U.S.A : Sage Publications, 2004.
- [9] TEODOROVIĆ, D. 2008. *Swarm intelligence systems for transportation engineering: Principles and applications.* Srbsko : Elsevier, 2008.
- [10] WEDDE, F. H., FAROOQ, M. and ZHANG, Y. 2004. *Beehive: An efficient faulttolerant routing algorithm inspired by honey bee behavior.* Berlín : Springer, 2004.
- [11] WONG, L. P. and CHONG, Ch. S. 2009. *An Efficient Bee Colony Optimization Algorithm for Traveling Salesman Problem using Frequency-based Pruning.* Wales : IEEE International Conference, 2009. ISBN: 978-1-4244-3759-7.

Ing. Marián Šedivý, PhD.
marian.sedivy@gmail.com
0907 305 014

Skúsenosť s prechodom od LMS Moodle ku cloudovému LMS iSmart

Experience with change from LMS Moodle to cloud LMS iSmart

Ján Papula

Abstract:

The evolution of information and communication technologies is influenced by trends, which should be perceived and reflected. This also applies to e-learning in support of higher education. Web-based services, such as Software as a Service or Cloud Computing, and also the development of mobile devices and mobile internet, and the development of modern format for distributing web content, are some of the trends that affect our everyday lives. This paper is devoted to reason and benefits obtained from the transition from the most widely used Open Source LMS Moodle to innovative new cloud solution iSmart.

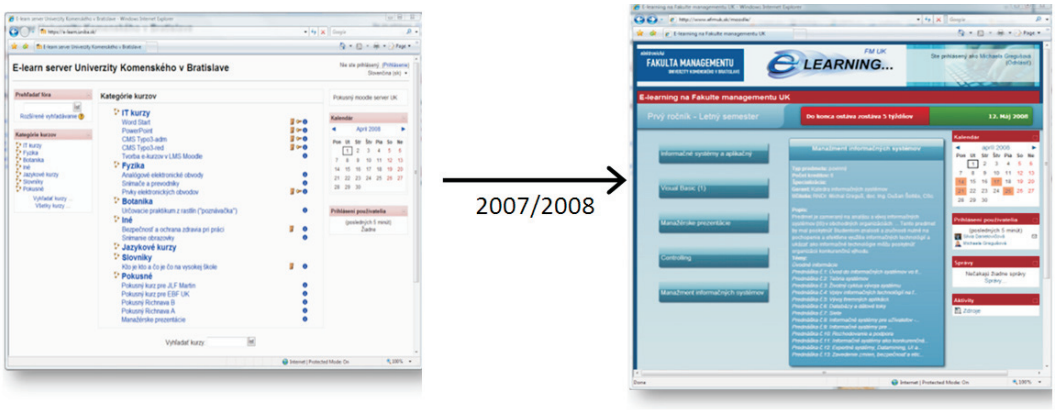
Keywords:

e-learning, cloud LMS, Learning Management System

ACM classification: I.2.6, K.3.1

1. Východiskový stav na Fakulte managementu Univerzity Komenského a požiadavky na inováciu e-learningového systému

E-learning na Fakulte managementu Univerzity Komenského v Bratislave sa buduje a postupne inovuje nad LMS Moodle už dlhšiu dobu, vlastné nasadenie inovovaného e-learningového systému prebehlo až počas akademického roka 2007/2008. V roku 2007 sme sa pri tejto inovácii zamerali na vytvorenie vlastnej nadstavby LMS Moodle na úrovni používateľského rozhrania študentského pohľadu, zefektívnenie a zjednodušenie prístupu používateľov aj výraznou redukciou funkcií a možností v učiteľskom prístupe do LMS Moodle. Používateľské rozhranie bolo vytvorené v technológii Flash, pričom viaceré prehľady známe z LMS Moodle boli prezentované v upravenej forme pre dosiahnutie ľahšieho prehľadu a lepšej orientácie v systéme, ako aj v kurze. Cieľom bolo aplikovať efektívnu formu distribúcie znalostí a účinnú podporu klasického vzdelávania, ktorá by spojila kvalitu poskytovaných informácií s možnosťami moderných informačno-komunikačných technológií. Snahou bolo nielen sprístupniť študentom informácie k samoštúdiu, ale aj naučiť ich sa učiť.[2] [4]



Obr. 1: *Zmena používateľského rozhrania pri LMS Moodle*

Už pri tejto implementácii boli požiadavky na vytvorenie moderného a efektívneho systému elektronickej podpory výučbového procesu. Sledovali sme trendy nie len v oblasti e-learningu, ale aj iných webových služieb, ako aj významné faktory vplývajúce na zmenu prístupu používateľov k využívaniu informačno-komunikačných technológií a elektronických zdrojov. [1] [5]

Aj keď sa v rámci testovania spokojnosti študentov v roku 2008 až 92 % študentov (z počtu 290) vyjadrilo, že zavedený e-learning je prínosný ako doplnok bežného vyučovania (z čoho 34 % označilo e-learning ako veľmi prínosný), ***celkovo pre nás LMS Moodle po rokoch skúseností predstavoval viac nedostatkov ako prínosov***, najmä z pohľadu ambície poskytovať študentom kvalitné moderné služby a podporiť touto formou väčšinu predmetov na fakulte.

Takýto dosiahnutý stav sme samozrejme nepovažovali za dostatočný, a to najmä pri sledovaní úrovne e-learningu na zahraničných univerzitách. Vzhľadom na vysoký počet študentov, ktorí sa každoročne zúčastňujú medzinárodných výmenných programov, je tak pre Fakultu managementu elektronické sprístupňovanie študijných podkladov k predmetom významným prvkom v zabezpečovaní služieb študentom.

Od spomínaného akademického roku 2007/2008 sme neustále monitorovali a vyhodnocovali spätnú väzbu od študentov, ktoré síce viedli k niekoľkým parciálnym inováciám, avšak po celkovom zhodnotení stavu sme pristúpili v roku 2012 k väčšej zmene.

Hlavné dôvody pre zmenu e-learningového systému:

- vysoká náročnosť na správu a údržbu open source LMS Moodle,
- potreba vyššej rýchlosti a stability systému,
- zjednodušenie a sprehľadnenie učiteľského prístupu do LMS,
- potreba prispôsobenia mobilným zariadeniam,
- doplnenie funkcií v rámci skvalitnenia poskytovaných služieb študentom,
- lepšie zosúladenie štruktúry LMS so štruktúrou a systémom výučbového procesu na fakulte.

1.1 Zhodnotenie skúseností z open source LMS Moodle – východiská pre potrebu inovácie

Hoci využívanie open source riešení v prvom momente evokuje pocit, že ide o bezplatné riešenia, realita často ukáže, že tomu tak nie je. Často sa pri implementácii akýchkoľvek informačných systémov totiž zabúda, že okrem obstarávacích nákladov vznikajú náklady v samotnom procese implementácie, ale aj náklady súvisiace s prevádzkou informačného systému a jeho potrebnou správu. V prípade open source systémov rastú so zložitosťou systému práve náklady na

implementáciu, ako aj na správu. Po viacročnej skúsenosti s týmto e-learningovým systémom sme identifikovali niekoľko skutočností, ktoré vplývajú na náročnosť prevádzky open source LMS Moodle:

a) Bezpečnosť systému

Správa informačných systémov, ktoré obsahujú osobné údaje používateľov si vyžadujú ich patričné zabezpečenie. Vzhľadom na to, že Moodle je open source systém vyvíjaný a rozvíjaný širokou komunitou viac či menej technicky zdatných ľudí, vzniká oprávnená obava o zabezpečenie systému. Pri monitorovaní množstva slabých miest systému zverejňovaných na webových stránkach Moodle komunity je aktuálna otázky ich priebežného zabezpečovania. Súčasne monitoring útokov na systém (napr. zapisovaním nežiaduceho obsahu do databázy) preukázal, že tieto obavy sú oprávnené. Najmä väčšie a rozsiahlejšie útoky na server počas vlny útokov skupiny Anonymous nás presvedčili, že stabilita a bezpečnosť systému, ako aj objektívna schopnosť ochrany osobných údajov používateľov je na veľmi nízkej úrovni. Okrem slabín z pohľadu zabezpečenia systému pre plnenie ustanovení Zákona o ochrane osobných údajov, tu zohráva svoju rolu aj schopnosť zabezpečiť obsah kurzu pred neželanými rozširovaním materiálov, ak si tvorca kurzu neželá sťahovanie a rozširovanie študijných podkladov mimo fakulty.

b) Aktualizácie systému

LMS Moodle ako systém, ako aj jednotlivé doplnkové komponenty prebiehajú neustálym inovačným procesom, t. j. v intervaloch sú uvoľňované aktualizované verzie tohto systému. Vzhľadom na to, že väčšina systémov na Fakulte managementu UK prebieha na platforme od Microsoftu, práca so systémom Linux, na ktorom bola inštalácia LMS Moodle predstavovala určité dodatočné náklady. Udržiavanie systému (aj obnovy systému po väčších hekerských útokoch) si tým pádom vyžadovalo väčšiu prácnosť. Spracovanie aktualizácií, ako aj testovanie zmien a kompatibility jednotlivých komponentov boli záťažou tak pre technickú správu, ako aj pre samotných učiteľov, ktorí boli hlavnými využívateľmi e-learningu. Zladenie týchto zmien a prípravy vzhľadom na harmonogram semestra si tak vyžiadalo neraz nárazovú prácu, ktorá samozrejme nevplývala pozitívne na učiteľov zabezpečujúcich elektronické kurzy. V prípade odbremenenia učiteľov by bolo potrebné na tieto aktivity zamestnať dodatočných administratívnych pracovníkov, čo s prihliadnutím aj na nárazovosť práce nie je efektívne.

c) Zložitosť databázového systému

LMS Moodle je systém, ktorý zabezpečuje všetky potrebné funkcie súvisiace s e-learningom, ale proces jeho vzniku, ale najmä aktualizácii a rozširovania vytvoril v tomto systéme vysokú komplikovanosť databázového systému s množstvom čiastkových tabuliek, s ktorými systém pracuje v jednotlivých funkciách. Táto zložitosť a značná neusporiadanosť v štruktúrach spôsobuje nie len nároky na porozumenie dátového modelu systému, ale najmä komplikuje situáciu pri vytváraní nadstavieb a rozšírení. Grafické používateľské rozhranie, ktorým sme študentom prezentovali podstatné informácie a efektívny pohľad na štruktúru kurzov ako aj ich obsah, si tak pre zobrazenie špecifických výstupov vyžadoval spracovanie dát z rôznych tabuliek databázy, čo výrazne predlžovalo odozvu systému a tým znižovalo hodnotu systému v očiach študentov, ktorí sú dnes zvyknutí na moderné informačné systémy s prehľadným a príťažlivým grafickým rozhraním a súčasne rýchlymi odozvami. Pri hodnotení spokojnosti s e-learningom sa v roku 2008 až 63 % študentov (z počtu 290) sťažovalo na rýchlosť systému a dlhé čakacie doby pre načítanie obrazoviek.

d) Vysoká zložitosť LMS Moodle pri vytváraní kurzu

Viaceré prieskumu a štúdie hovoria o jednoduchosti LMS Moodle a učitelia, ktorí s ním pracujú sa vyjadrujú vcelku pozitívne o funkcionalitách ako aj náročnosti. Avšak keď zvážime nízke percento učiteľov, ktorí najmä v netechnických odboroch pracujú s týmto systémom, či nízke percento predmetov pokrytých e-learningom na jednotlivých fakultách na Slovensku, tak

musíme dôjsť k záveru, že niekde je vážny problém. Súčasne, ak prihliadneme na fakt, že kurzy, ktoré sú ponúkané pre získanie vedomostí a zručností s LMS Moodle čítajú často niekoľko desiatok hodín štúdia, je zrejmé že práca so systémom vôbec nie je tak intuitívna a jednoduchá. V Slovenskom vysokoškolskom prostredí, kde sa od učiteľov okrem výučby a vedenia záverečných prác vyžaduje aj tvorivá a vedecká práca, sú požiadavky na technické zručnosti a znalosti v programovaní HTML stránok vo veľkej miere nereálne. Nehovoriac o zložitej administrácii systému pri každo-semestrovom nastavovaní kurzov a správe študentov na predmetoch. Ak si pritom uvedomíme prínosy súvisiace s využívaním e-learningu (či už ako doplnok prezenčnej formy štúdia, alebo ako priamo nástroj dištančného štúdia), je veľkou škodou, ak bariérou má byť technická zložitosť a z toho plynúca demotivácia tvorcov kurzov.

1.2 Trendy, ktoré ovplyvňujú využívanie informačno-komunikačných technológií

Pri hľadaní riešenia, ako zabezpečiť moderný a perspektívny systém elektronickej podpory vzdelávania, je potrebné zohľadniť trendy, ktoré súvisia s využívaním informačno-komunikačných technológií a s trendmi vysokoškolského vzdelávania.

Z nášho pohľadu rozhodujúce trendy, ktoré je potrebné pri budovaní, alebo inovovaní e-learningu zohľadniť, sú:

- Podpora mobility v IT a mobilných zariadení;
- Využitie výhod cloud computing-u;
- Zosúladenie e-learningu s požiadavkami vnútorného systému zabezpečovania kvality vzdelávania na báze ESG.

Využívanie počítačových aplikácií, či internetových služieb už dávno nie je fixované na miesto, kde je postavený stolový osobný počítač. Používatelia si už zvykli na mobilitu a na prístup k informačno-komunikačným technológiám kedykoľvek a kdekoľvek.

Práve mobilita je jedným z kľúčových trendov pri využívaní a ďalšom rozvoji e-learningu, a to tak v akademickom, ako i komerčnom prostredí. Zníženie cien notebookov v poslednom desaťročí, ale tiež nástup inovácií vo forme prenosnejších netbookov či ultrabookov odpútalo používateľov od stolových počítačov a umožnilo im preniesť si pracovné či študijné prostredie doslova kamkoľvek. Tento záujem o prácu a štúdium kedykoľvek a kdekoľvek podľa potreby dokazuje aj výrazný posun v predajnosti od tradičných stolných počítačov práve k mobilným zariadeniam, aktuálne k tabletom. Aj v dnešnom zrýchlenom komerčnom pracovnom svete je pre mnohých pracovníkov vzácne, nájsť si čas na nevyhnutné štúdium za pracovným stolom počas pracovnej doby. Pracovníci na všetkých úrovniach preto veľmi vítajú nové technológie, ktoré im pomáhajú kdekoľvek kedykoľvek rozvíjať svoje odborné zručnosti[3].

Na využívanie mobilných zariadení v e-learningu sa upozorňovalo už v minulosti. Koncept mLearningu nie je nový, avšak k jeho skutočnému rozvoju nedošlo v takej miere, ako predpokladali jeho presadzovatelia. Jednou z príčin môže byť aj to, že pri mobilných zariadeniach sa uvažovalo s inou formou distribúcie obsahu (SMS, MMS, hlasové odkazy), ako aj s potrebou prispôsobeniu sa menším rozlíšeniam obrazu. Mobilné zariadenia tak boli vnímané ako iný typ zariadenia pre distribúciu iného, viac personalizovaného e-learningového obsahu. Zabezpečiť realizáciu paralelného alebo doplnkového systému k už zavedenému e-learningu však nie je častokrát ani kapacitne, ani ekonomicky zvládnuteľné. Nástup nových moderných mobilných zariadení, ktoré odštartovalo uvedenie 3G verzie mobilného telefónu iPhone v júni 2008 a následne v apríli roku 2010 predstavenie tabletu iPad ukazuje, že používatelia nechcú na mobilných zariadeniach konzumovať redukovaný a upravovaný obsah, ale práve naopak, majú záujem o plnohodnotné prehliadanie obsahu v porovnateľnej vizuálnej a obsahovej kvalite ako na stolnom počítači, či notebooku. Rovnako rozvoj mobilného internetu prináša nové možnosti a odstraňuje

minulé limity týchto zariadení pri konzumovaní elektronického obsahu. Nie len rozvoj bezdrôtového WiFi pripojenia, ale najmä nárast pokrytia 3G mobilným internetom umožňuje používateľom moderných mobilných zariadení sťahovať a prehliadať elektronický obsah vo vysokej kvalite takmer bez obmedzení. Ako uvádza Bob Little [3], hoci bol mLearning v minulosti len čiastkovým a doplnkovým prúdom e-learningu, dnes sa mobilné zariadenia stávajú hlavným prostriedkom pre distribúciu vzdelávacieho obsahu.

Je preto zaujímavé, že aj v čase, keď mobilné zariadenia disponujú bežne rozlíšením obrazovky na úrovni stolných počítačov a notebookov (ba dokonca často vyšším) a súčasne disponujú rýchlym pripojením k internetu, potreba prispôsobenia obsahu mobilným zariadeniam sa stále nachádza v aktuálnych výskumných štúdiách[6].

Ďalšie trendy je vidieť v zvyklostiach konzumovania elektronického obsahu používateľmi mobilných zariadení. Stále menej elektronického obsahu je sťahovaných a ukladaných do týchto mobilných zariadení, obsah je prehliadaný a konzumovaný priamo v prostredí webových služieb. Aj relatívne nízka veľkosť pamäte na tabletoch a smartfónoch je reflexiou tento trend. Podporujú to aj stále rozširujúce sa cloudové služby, kedy k aplikáciám ale aj celým platformám pristupujeme cez internet ako k službám, bez potreby inštalácie dodatočných komponentov do osobného zariadenia. Týka sa to aj prehliadania multimediálnych súborov, čo viedlo zvýšenému záujmu o štandard HTML5. V oblasti webových služieb je dobrým príkladom smerovania podpory mobility v informačno-komunikačných technológiách využívanie e-mailových služieb (napr. Gmail, Outlook), video či audio portálov (napr. YouTube, SoundCloud), webových úložísk (napr. Dropbox, SkyDrive), alebo celých aplikácií (napr. Office 365, Google Docs). [7]

Tento trend je zjavný aj v oblasti vzdelávania. Používatelia sa už dnes bežne stretávajú s modernými vzdelávacími systémami, či portálmi, ktoré im ponúkajú jednoduché používateľské rozhranie, ako aj podporu HTML5. Dôkazom je rozšírenie a vysoká popularita otvorených on-line vzdelávacích portálov (MOOC – Massive Open Online Courses). Nie všetci prevádzkovatelia e-learningových portálov, a to najmä vo vysokoškolskom prostredí, na to však zareagovali. Sťahovanie a ukladanie objemných multimediálnych súborov, či obyčajných dokumentov (.doc/.ppt/.pdf), ktoré neraz zbytočne zahlcujú pamäťový priestor v mobilných zariadeniach (najmä ak sa opakovane sťahujú pri opakovaných návštevách toho istého kurzu) by mali nahradiť moderné HTML5 objekty. Zložité štruktúrované portály založené na princípe "čím viac, tým lepšie", by mali nahradiť jednoduché a intuitívne prostredia. Práve veľká obľuba MOOC portálov Coursera či Khan Academy vytvárajú veľkú kvalitatívnu priepasť voči tradičným LMS v prostredí fakúlt.

Podstata potrebnej inovácie však nie je v obsahu, ale vo forme distribúcie vzdelávacieho obsahu.

2. Hlavné funkcie cloud LMS iSmart a skúsenosti s prechodom z LMS Moodle

iSmart je e-learningový systém postavený na princípe manažovaného prostredia – t. j. prostredia, ktoré je plne pod kontrolou a súčasne ktoré dokáže získavať a prezentovať podnety potrebné pre inovácie. iSmart má vlastnú e-learningovú SMART metodiku, ktorá je v stručnosti založená na:

Specific – každý objekt má svoj cieľ a prínos (koncept Mastery learning);

Measurable – štúdium je priebežne vyhodnocované a monitorované;

Attainable – objekty by mali na seba znalostne nadväzovať (koncept Bloom's Taxonomy);

Relevant – priebeh vzdelávania napĺňa ciele a je prínosom personálnom rozvoji;

Timed – objekty aj kurz sú časovo nastaviteľné v súlade so vzdelávacím plánom.



Obr. 2: *Jednoduché prostredie prispôsobené aj pre dotykové zariadenia*

Vnútorná logika kurzov založená na podmienenej postupnosti v štúdiu umožňuje sledovať progres a tempo štúdia, rovnako ako plnenie nastavených povinností pre prípadné využitie pre priebežné hodnotenie štúdia. Prepojenie kurzov do systému znalostných máp na princípe tematických prerekvizít zas umožňuje vyhodnocovať študijné profily a tiež viesť študujúcich pri napĺňaní svojich študijných cieľov. Vyhodnocovanie štúdia tak nie je len na úrovni elementárnych štatistických reportov, ale umožňuje vyhodnocovať stav a správanie sa študujúcich pre manažment vzdelávacieho procesu.

Ako spätná väzba tu funguje hodnotenie elektronických kurzov študentmi. Ide pritom o významný infomačno-kontrolný nástroj pre proces neustáleho zlepšovania (podľa Demingovho cyklu P-D-C-A). V každom kurze je možné nastaviť spustenie hodnotenia študentmi (hodnotenie sa spustí len v predmetoch, kde to učiteľ sám zaktivuje). Keďže hodnotenie sa v čase môže meniť, je to dobrý feed-back pre učiteľa aby vedel z ktorej oblasti si študenti žiadajú viac alebo inak spracovaných podkladov.

Pre potrebu fakulty alebo vysokej školy je k dispozícii automatizovaný ranking kurzov, ktorý vyhodnocuje stupňa vyspelosti “Maturity Level” kurzov podľa východiskovej metodiky SMART (v prípade potreby s možnosťou úpravy váh a kritérií hodnotenie podľa vnútorných pravidiel e-learningu fakulty).

Fakulta alebo vysoká škola tak vie vyhodnocovať na akej úrovni sú spracované jednotlivé kurzy. Taktiež robiť analýzy (na úrovni stupňov štúdia, ako aj študijných odborov), z pohľadu úrovne kvality elektronickej podpory vzdelávacieho procesu

iSmart je tak e-learningový systém pre dištančnú alebo kombinovanú metódu štúdia v manažovanom prostredí, ktorý ***umožňuje vysokej škole vyhodnocovanie štúdia aj pre potreby vnútorného systému zabezpečovania kvality vysokoškolského vzdelávania***. Hlavné funkcie systému iSmart sú:

- iSmart je cloudový systém, t. j. využíva sa ako služba, bez potreby inštalácie a systémovej administrácie na strane vysokej školy;
- iSmart je integrovateľný cez jednotné prihlasovanie (single sign-on) s intranetovým systémom na strane vysokej školy, napr. Microsoft Sharepoint;
- XHTML/HTML5 podpora pre plnohodnotné prehliadanie cez mobilné zariadenia (Smartfóny, Tablety);
- SCORM kompatibilita umožňuje vkladať objekty vytvorené modernými autoringovými nástrojmi, resp. vzdelávacie objekty z iných e-learningových systémov (napr. z Moodle);
- štruktúra kurzov a vzdelávacích objektov umožňuje nastaviť vnútornú logiku vzdelávania pre podporu obsahového prepojenia, ako aj zabezpečenie želanej nadväznosti podľa požiadaviek vzdelávacieho plánu organizácie;

- výstupy zo vzdelávania formou reportov, resp. automatizovane generovaných certifikátov (na základe kritérií nastavených osvedčeným lektorom) s možnosťou ich priameho zasielania na príslušné kontakty v organizácii (napr. na personálne oddelenie), alebo do podnikového workflow systému.

Z pohľadu vnútorného systému zabezpečovania kvality výučbového procesu prináša LMS iSmart niekoľko podporných nástrojov:

- umožňuje spúšťať vstupné a výstupné testy na úrovni kurzu s automatickým vyhodnocovaním výsledkov, ako aj progresu v štúdiu samostatne za jednotlivé študijné skupiny;
- umožňuje vyhodnocovať plnenie študijných povinností študentov v prostredí e-learningu, a to po jednotlivých študijných skupinách;
- umožňuje export štatistík pre ďalšie spracovanie napr. vo fakultnom Business Intelligence nástroji;
- umožňuje hodnotiť úroveň vyspelosti (Maturity Level) elektronických kurzov, a tým z pohľadu fakulty rýchlo vyhodnocovať úroveň e-learningovej podpory, najmä v prípade používanej dištančnej alebo kombinovanej metódy vzdelávania;
- umožňuje v termínoch začatia skúškového obdobia automaticky spúšťať hodnotiace dotazníky (pre vyjadrenie spätnej väzby študentov na predmet) samostatne pre každý kurz, a ich následné anonymné spracovanie a vyhodnocovanie.

Prvé testovanie nového cloudového e-learningového systému iSmart prebehlo v roku 2012. V rámci tohto testovania bola súčasne zisťovaná aj miera napĺňania zámerov ako aj prínosy tohto nového LMS.

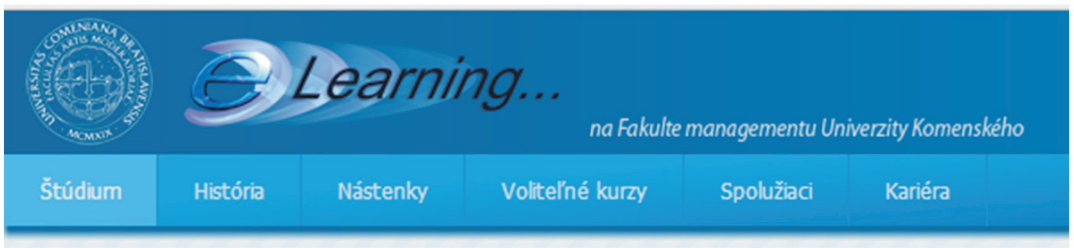
Na fakulte prebehlo v úvode používania školenie učiteľov, pričom čas potrebný na plné zvládnutie práce s e-learningom iSmart bol stanovený na 2 hodiny. Vzhľadom na to, že iSmart podporuje SCORM štandard, presun kurzov, ktoré prebiehali v LMS Moodle bol jednoduchý a bezproblémový.

Pri využívaní moderných autoringových nástrojov (najmä od spoločnosti iSpringsolutions, Articulate, FlippingBook) sa práca učiteľa sústreďuje takmer výhradne na prácu s nástrojmi MS Office, resp. s jednoduchým prepájaním na externé zdroje (napr. YouTube). Vkladanie objektov do kurzu je jednoduché ako pripájanie príloh k emailom. Snaha je o vkladanie obsahu kompatibilného s HTML5.



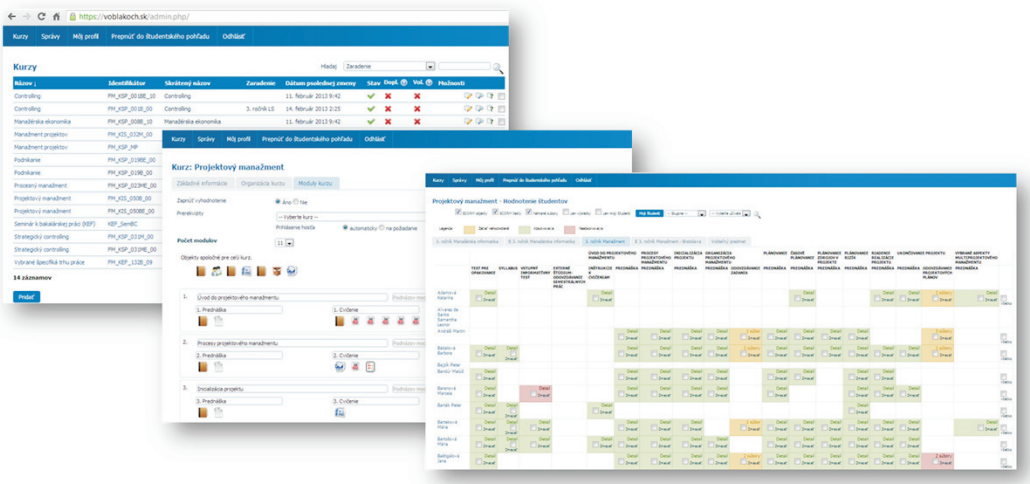
Obr. 3: *Jednoduchá migrácia kurzov z LMS Moodle*

Fakulta managementu dostala prístupové práva pre využívanie cloudového LMS ako služby. Bolo vytvorené jedno konto pre administrátora e-learningu zo strany fakulty a celá administrácia je tak plne v rukách fakulty. Administrátor sa stará o správu používateľov, kurzov a študijných skupín, pričom používatelia, v štruktúre podľa študijných skupín, ako aj predmety (spolu s priradením učiteľov k predmetom), boli do systému importované z upravených exportov z Akademického Informačného Systému (AIS2) formou štruktúrovaných .csv súborov. Učiteľ sa už o administrovanie nestará, venuje svoju pozornosť len správe svojich kurzov alebo komunikácii so študentmi. Z toho pohľadu bolo jednoznačne zabezpečené, že každý používateľ (tak študent ako aj učiteľ) po prihlásení vidí len kurzy, ktoré mu patria. U študentov sa kurzy aktuálneho semestra zobrazujú v základnej záložke štúdium, zabsolvované kurzy sú presúvané do záložky história (pričom študent tu vidí vždy aktuálnu verziu kurzov, t. j. v prípade vstupu do kurzu s odstupom niekoľkých rokov, študent vidí aktualizovaný kurz, a keďže pristupuje k nemu ako hosť, učiteľ ani iní študenti jeho aktivitu priamo neregistrujú). Voliteľné kurzy, ktoré sú ponúkané príslušným študijným skupinám podľa študijného programu si študent vyberá zo záložky Voliteľné kurzy. [7]



Obr. 4: *Ponuka záložiek v študentskom pohľade*

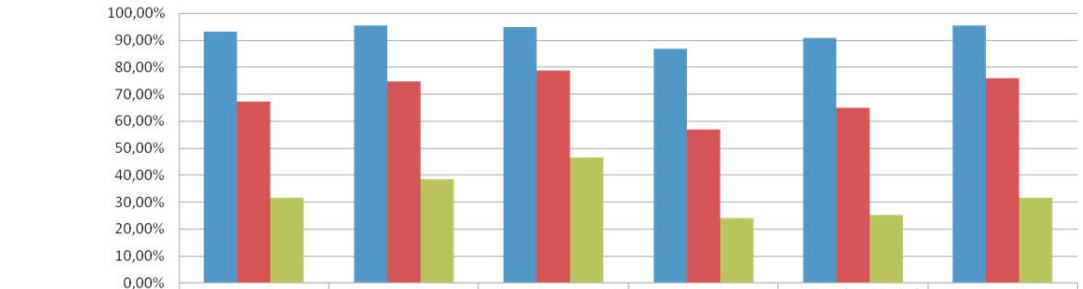
Súčasne iSmart LMS umožňuje aj adresnú komunikáciu so študentmi. Umožňuje to systém správ a násteniek, keď každá skupina a každý predmet má samostatnú nástenku (ktoré sa študentovi agregujú do jednej obrazovky podľa študijných skupín v ktorých študuje ako aj súboru predmetov, ktoré má aktuálne zapísané). Učiteľ si vyberá do ktorých skupín, resp. kurzov chce poslať správu aj s možnou prílohou. Druhým vizualizačným prostriedkom je kalendár s integrovaným zoznamom úloh, pričom do kalendára sa študentovi automaticky načítavajú všetky dôležité udalosti z kurzov, ktoré má aktívne, ako aj správy, ktoré učitelia pridelia k niektorému dátumu. [7]



Obr. 5: *Hlavné obrazovky učiteľského pohľadu*

Po spustení pilotného testovania sú dosiahnuté nasledovné priebežné výsledky:

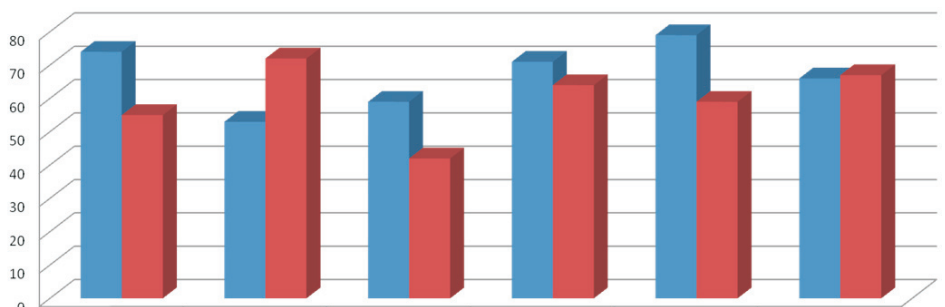
- Počet kurzov do ktorých učitelia začínajú vkladať elektronický obsah sa z roku 2011 (18 kurzov) zvýšil po letný semester 2013 na 46 – t. j. nárast o 155 %
- Pri dotazníkovom prieskume, ktorého sa zúčastnilo 174 študentov boli získané nasledovné hodnotenia inovácie e-learningu prechodom na nový cloudový LMS iSmart:



	Rýchlosť načítania stránok systému	Prehľadnosť kurzov a systému	Dizajn systému (prvkov, stránok)	Funkcionalita systému	Možnosť prehliadať na rôznych platformách / zariadeniach	Celkové zhodnotenie inovácie
Pozitívne	93,10%	95,40%	94,83%	86,78%	90,80%	95,40%
z toho: Dobrá inovácia	67,24%	74,71%	78,89%	56,90%	64,94%	75,86%
z toho: Výrazný progres	31,61%	38,51%	46,55%	24,14%	25,29%	31,61%

Obr. 6: *Hodnotenie inovácie e-learningu študentmi FMUK*

Súčasne bol v závere letného semestra 2013 uskutočnený prieskum na e-learningom pokrytom predmete 3. ročníka bakalárskeho stupňa štúdia – Controlling, ktorého sa zúčastnilo 122 študentov tohto predmetu (58 študentov denného štúdia a 64 študentov externého štúdi) s nasledovnými výsledkami:



	Možnosť opakovať si pred skúškou	Možnosť priebežne si študovať materiály a podklady	Väčšia ponuka študijných podkladov v porovnaní s predmetmi bez e-learningu.	Nie je potrebné toľko písať poznámok na hodinách	Nie je potrebné nakupovať literatúru v hodnote 20-50EUR	Možnosť priebežne sa testovať (za účelom opakovania)
Denní	74	53	59	71	79	66
Externí	55	72	42	64	59	67

Obr. 7: *Prieskum prínosov e-learningu študentmi FMUK na predmete Controlling*

Celkovo je možné považovať prechod na nový LMS iSmart ako správny smer rozvoja e-learningu na Fakulte managementu. V prvom roku nasadenia prevládajú pozitívne hodnotenia tak zo strany študentov, ako aj učiteľov, ktorí podporili svoje predmety aj elektronickou formou.

Súčasne s touto inováciou Fakulta managementu prešla v roku 2012 aj na cloudový e-mailový systém, ktorý je súčasťou Microsoft Office 365. Postupne sa nasadzuje aj intranetové riešenie Microsoft Sharepoint. Do najbližšieho obdobia nás tak čaká nie len ďalší rozvoj e-learningových kurzov, ale tiež prepojenie LMS iSmart pod jednotné prihlasovanie s MS

Sharepoint, ako aj spustenie doplnkových modulov LMS iSmart, ktoré zatiaľ neboli na Fakulte managementu uvedené do prevádzky. Pôjde najmä o elektronické virtuálne kariérne centrum a obchod s kurzami, v ktorom budú môcť študenti pristupovať k externým e-learningovým kurzom, resp,. v rámci ktorého budú môcť aj študenti zdieľať svoje znalosti prostredníctvom vlastných vytvorených kurzov. Od týchto doplnkových služieb si sľubujeme ďalšie zatraktívnenie zvýšenie návštevnosti e-learningového systému.

Použitá literatúra:

[1] BEŇOVÁ, E. – HLIVÁROVÁ, N. – GREGUŠ. M. 2005. Distance Education as e-Service to Support Higher Quality of Education, In: Global Business and Economic Development. – Montclair : Montclair State University, s. 670-674, ISBN 978-0-9747415-4-3.

[2] BEŇOVÁ, E. – HLIVÁROVÁ, N. – GREGUŠ. M. – PAPULA, J. 2008. Elektronické vzdelávanie na Fakulte managementu, In: Proceedings of the information day on the EU funded IWebCare project and workshop on e-Europe (CD-ROM). – Bratislava : FM UK, s. 38-42, ISBN 978-80-223-2619-3.

[3] LITTLE, B. 2012. Effective and efficient mobile learning: issues and tips for developers,INDUSTRIAL AND COMMERCIAL TRAINING Journal VOL. 44 NO. 7, Emerald Group Publishing Limited, s. 402-407, ISSN 0019-7858

[4] PAPULA, J. – GREGUŠ. M. 2008. Koncepcia krátkeho prístupu v e-Learningu na Fakulte managementu, In: eLearn 2008. – Žilina : Žilinská univerzita, s. 193-199, ISBN 978-80-8070-838-2.

[5] PAPULA, J. 2009. E-learning 2.0 čo prináša a aké je jeho miesto v znalostnom manažmente podniku, eFocus. Roč. 9, č. 2, s. 22-26 ISSN 1336-1805.

[6] QUINN, C. N. 2011. Mobile Learning: Landscape and Trends, The eLearning Guild Research 2011

[7] REHÁK, J. 2013. Zmena k modernému cloudovému LMS. Dnešné trendy inovácií. DTI v Dubnici nad Váhom, s. 190-199, ISBN 978-80-89400-60-7

doc. Ing. Ján Papula, PhD.
Fakulta managementu
Univerzita Komenského v Bratislave
Odbojárov 10
820 05 Bratislava
jan.papula@fm.uniba.sk



Slovak Society for Operations Research

Slovak Society for Operations Research (Slovenská spoločnosť pre operačný výskum) was established in 1993 after splitting Czechoslovak Society for Operations Research into Slovak and Czech Society.

The aims of the Slovak Society for Operations Research are:

(a) to support the development of operations research as a scientific discipline. For this purpose the Society organizes meetings, conferences and seminars and supports publishing scientific and applied results in operations research. It organizes scientific study trips of its members at institutions, the work of which is oriented to the theoretical development or applications of operations research both in the Slovak republic and abroad;

(b) to support educational and enlightenment activities devoted to the area of operations research. For this purpose, The Society organizes lectures and panel discussions for people working in the management sphere of industry, commerce and agriculture, in the central management institutions and education with the aim to inform them about the possibilities of applications of operations research in the spheres of their professional activity. It carries out the advisory activity concerning the possibilities of gaining qualification in operations research in all public and private institutions. The Society is the custodian of the standardization of operations research terminology in Slovak languages;

(c) to inform about useful applications of operations research methods. For this purpose the Society provides free of charge orienting consultations for private firms and public institutions. The consultations are provided by the members of the Society who have the best professional abilities for the successful solution of the given problem. The Society supports the consultation activity in operations research, which is carried out on a commercial basis and gives objective information about its level;

(d) to organize and coordinate contacts with partner institutions of similar type abroad. For this purpose it supports in the frame of its budget the participation of its members at specialized conferences abroad and organizes the working trips of foreign operations research specialists in the Slovak republic.

Nowadays represents of Slovak Society for Operations Research **president** doc. Mgr. Juraj Pekár, PhD.

The Slovak Society for Operations Research organized:

International workshop QUANTITATIVE METHODS IN SCIENCE AND PRAXIS

This workshop is organized every second year for the members of the Slovak Society for Operations Research and representatives from Czech Republic (namely University of Economics

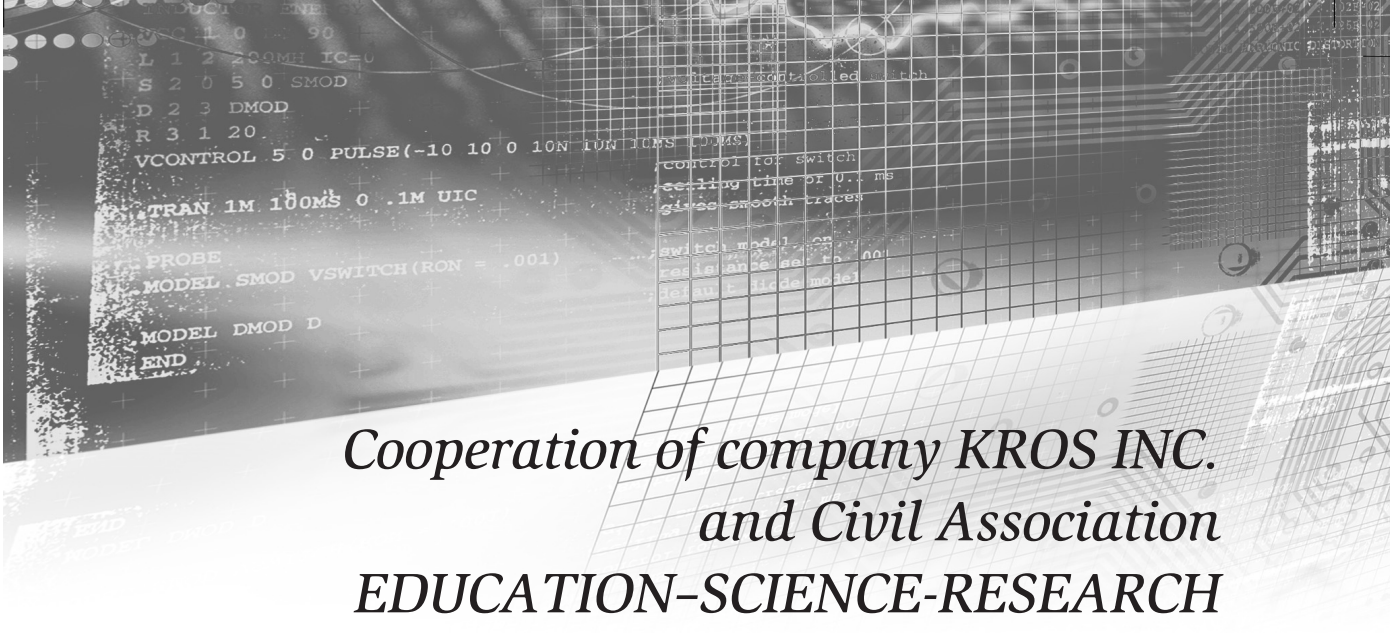
Prague and Technical University in Ostrava). We deal there with study programs oriented on the operations research for the doctoral, graduated and under graduated studies. It is important to note, that the basic ideas of the lectures and seminars aimed at the operations research were established at the Department of economic and mathematical calculations in year 1961, later, Department of Operations Research and Econometrics the University of Economics in Bratislava. Professors of the Slovak Society for Operations Research are in principle the guarantors of the lectures and seminars focused on the operations research offered in other universities in Slovakia. Subjects of operations research are delivered besides University of Economics in Bratislava also in Comenius University in Bratislava, Slovak Agricultural University in Nitra and University of Žilina and partly in Technical universities in Bratislava and Košice. Subjects of operations research are modified according to specific study programs of the above-mentioned universities.

International conferences QUANTITATIVE METHODS IN ECONOMICS (Multiple Criteria Decision Making).

This conference is organized every second year. Presentations are focused on the methodology and construction of the models and applications of the methods of operations research and econometrics. We would like to stress that each conference is especially devoted to the actual problems of economic policy on macro and micro level solved by models and methods of operations research and econometrics. **Proceedings** of all international conferences QUANTITATIVE METHODS IN ECONOMICS are on the page: http://www.fhi.sk/en/katedry-fakulty/kove/ssov/papers/.

The next XVII international conference QUANTITATIVE METHODS IN ECONOMICS will organized under Slovak Society for Operations Research in 2014.

Slovak Society for Operations Research
KOVE FHI EU, Dolnozemská 1/b
852 35 Bratislava, Slovakia
e-mail: ssov@dec.euba.sk



Cooperation of company KROS INC. and Civil Association EDUCATION–SCIENCE-RESEARCH

Company KROS INC. (*Company*) decided in 2012 to establish cooperation with CivilAssociation EDUCATION–SCIENCE-RESEARCH(*ESR*), which supports development of education, science and research by using modern technologies and resolving scientific research projects. Subjects of this cooperation are following:

- education,
- creation of common courses,
- support of promotional activities.

By discussions it was approved that *ESR* will be with the support of the *Company* authorized in following activities:

- lectures / courses of Energy economics,
- lectures / courses at high schools aimed to promote products of the company KROS INC.

Company helps Slovak high schools with teaching of economic subjects, accounting and construction. It realizes that it is necessary for teaching of young people to be supported by newest technologies and quality software. Therefore, the *Company* carried out an extensive project *Support of schools* and provides:

- free school versions of programs ALFA plus, OLYMP, OMEGA, CENKROS plus,
- free upgrade,
- 2 free pieces of user manuals for each program,
- free use of services HOT-LINE,
- free e-mail newsletter of the project,
- participation at all types of training with 60 % discount,
- other publications and services with 60 %.

Upon acceptance of specific conditions also provides:

- 15 free pieces of user manuals for each program,
- freetraining of pedagogues in the office of the *Company*,
- free training of pedagogues in the school area (reimbursement of only travel costs),
- free help with installation of PC programs in the school area (reimbursement of only travel costs),

- free regular training for pedagogues,
- freeaward certificates of the *Company*for students.

School version is the full version of the program, which is provided for educational facilities and educational organizations in Slovakia exclusively for teaching. School versions of programs ALFA plus, OLYMP and OMEGA does not allow to enter dates in months July and August, i.e. the program can be used also in these months, however with different dates. School version of CENKROS plus does not allow to export/import contracts and print reports in MS Excel format and print reports contain information that they have been processed in a school version. Every type of school facilities (primary, secondary, Universities) have right for school version regardless of the type of founder (with legal personality, private, public, etc.) and all types of educational organizations (academies etc.). School versions of programs are provided in accordance with applicable policy of the company. Further information for participation on the project are available on http://www.kros.sk/skolskeverzie#Projekt.

Since September2012 the *Company*prepares seminars for pedagogues of economic schools. Seminars are designed for pedagogues, who participated in program of continuing education, or teach by economic programs of the *Company*.

Company KROS INC. is trying to cooperate with the Civil Association EDUCATION-–SCIENCE-RESEARCHon a professional level, which can be beneficial for all involved in education.

Ing. Radoslav Králik
KROS INC., economic and buildingsoftware
A. Rudnaya 21, 010 01 Žilina
kralik@kros.sk

Ing. Janette Brixová
Civil Association ESR
Andrusovova 5, 851 01 Bratislava
janette.brixova@gmail.com



Informatics 2013
The 12th International Scientific Conference

Spišská Nová Ves, Slovak Republic, 5th–7th November, 2013

GENERAL CHAIR

Liberios Vokorokos, Dean of Faculty of Electrical Engineering and Informatics, Technical University of Košice (SK)

HONORARY CHAIR

Ivan Plander, Slovak Society for Applied Cybernetics and Informatics (SK)

The 12-th International Conference Informatics'13 is a biennial international forum which gives room for scientists, experts in computer science and professionals in new emerging fields of informatics to present original research results, to share experience and to exchange ideas about transferring theoretical concepts into real life. The conference provides a unique opportunity for establishing and maintaining professional relationships of widely recognised scientists between each another in specific areas of computer science.

CONFERENCE TOPICS ARE AS FOLLOWS:

1. Computer Architectures
2. Computer Networks
3. Theoretical Informatics
4. Programming Paradigms, Programming Languages
5. Software Engineering
6. Distributed Systems
7. Computer Graphics and Virtual Reality
8. Artificial Intelligence
9. Knowledge Management
10. Information System Research
11. Applied Informatics and Simulation
12. Informatization of self-government in the Information Society

IMPORTANT DATES

Abstract submission extended to:	July 15th 2013
Full paper submission:	September 1st 2013
Notification of acceptance:	September 30th 2013
Registration, payment, conference program:	October 10th 2013
Conference:	November 5th-7th 2013

All informations about conference can be found at conference web site http://informatics.kpi.fei.tuke.sk.



Milan Šujanský,
Technical University of Košice, Košice, Slovakia
E-mail: Milan.Sujansky@tuke.sk
Phone: +421 55 602 2119

Instructions for authors

- Articles submitted for publication will be accepted only as Word document (.doc or .docx) sent to e-mail address of managing editor (vvv.esr@gmail.com).
- Range: Articles and papers in 20 standard pages, essays in 10 standard pages, reviews and gleanings in 5 standard pages (on one page A4 is possible to type 1800 characters including spaces)
- Text should be written as plain text (text font Times New Roman, size 12, without hyphenation and justification).
- If you need you can use bold, italic or underline
- If you use notes in the text, place the note on the same page.
- Quote following general rules (STN ISO 690)
- The author is responsible for the formal and scientific side of the article.
- In our journal is not possible to public the article already published in other journal.
- The editorial board reserves the right to edit the headline of the contribution, to make stylistic, grammar, language and graphical emendations and decide on its inclusion into specific numbers, and rubrics
- The editorial board reserves the right to refuse the contribution.
- Structure of the paper as follow

TITLE OF THE ARTICLE in ENGLISH

Title of the article in the other language

First and last name without academical degree; in the case of multiple authors, separate names with commas

Abstract in English (6-10 lines)
Key words in English (3-7 key words)
Key words in other language (optional)

The article
If you want the pictures to be nice and readable, add the bigger and better quality pictures at the end of the articles with the same numbering as in the article.
Pictures, tables, math formulas will be centered. Please use Fig. 1, Tab. 1,..to label the objects in the article.

Citations and References
In our journal we prefer to quote the citations by author (not by the reference number), so please try to follow this style (Editors, 2011).

Editors (2011).*How to create an article to publish it in ITA*. Retrieved July 19, 2011, from http://www.paneurouni.com/sk/fakulty/faculta-informatiky/bratislava-slovakia/o-fakulte/

About Author(s) brief information about author (optional)
(Add name with academical degrees)
Contacts
Name of the author with titles
Address of the institute
Email address: