

**Žilinská univerzita v Žiline**  
**Spoločnosť pre otvorené informačné technológie**

**OTVORENÝ SOFTVÉR VO VZDELÁVANÍ,  
VÝSKUME A V IT RIEŠENIACH**



**Zborník príspevkov medzinárodnej konferencie  
OSSConf 2015**

**1.–3. júla 2015  
Žilina, Slovensko**

## Otvorený softvér vo vzdelávaní, výskume a v IT riešeniach

1.–3. júla 2015, Žilina, Slovensko

### Vedeckí garanti konferencie:

prof. Ing. Karol Matiaško, PhD., Žilinská univerzita, Žilina

prof. Ing. Miloš Šrámek, PhD., Austrian Academy of Sciences, Wien (AUT); SOIT, Bratislava

### Vedecký a programový výbor:

prof. Ing. Miloš Šrámek, PhD., Austrian Academy of Sciences, Wien (AUT) – **predseda**

prof. Oleg Černojarov, DrSc., Moscow Power Engineering Institute (RU)

doc. Ing. Karol Grondžák, PhD., Žilinská univerzita, SOIT, Žilina

doc. RNDr. Štefan Peško, CSc., Žilinská univerzita, SOIT, Žilina

doc. Ing. Jiří Rybička, Dr., MZLU Brno (CZ)

RNDr. Rudolf Blaško, PhD., Žilinská univerzita, SOIT, Žilina

Mgr. Peter Czimmermann, PhD., Žilinská univerzita, SOIT, Žilina

Mgr. Michal Kaukič, CSc., Žilinská univerzita, SOIT, Žilina

RNDr. Aleš Kozubík, PhD., Žilinská univerzita, SOIT, Žilina

Dr inż. Remigiusz Olejnik, Zachodniopomorski Uniwersytet Technologiczny w Szczecinie (PL)

Ing. Pavel Stříž, Ph.D., Nakladatelství Martin Stříž, Bučovice (CZ)

RNDr. Ladislav Ševčovič, PhD., FEI, Technická Univerzita Košice, SOIT, Košice

Ing. Jiří Eischman, Red Hat CZ, SOIT, Brno (CZ)

### Organizačný výbor:

Aleš Kozubík, Žilinská univerzita, SOIT, Žilina – **predseda**

Rudolf Blaško, Žilinská univerzita, SOIT, Žilina

Michal Kaukič, Žilinská univerzita, SOIT, Žilina

Michal Chovanec, Žilinská univerzita, SOIT, Žilina

Tomáš Majer, Žilinská univerzita, SOIT, Žilina

Miloslav Ofúkaný, GeoCommunity, SOIT, Bratislava

Pavel Stříž, Nakladatelství Martin Stříž, Bučovice

Peter Štrba, Gymnázium Mikuláša Galandu, SOIT, Turčianske Teplice

Vydavateľ: Žilinská univerzita v Žiline

ISBN 978-80-970457-7-7

---

Copyright © 2015 autori príspevkov

Ktokoľvek má dovolenie vyhotoviť alebo distribuovať doslovný opis tohoto dokumentu alebo jeho časti akýmkoľvek médiom za predpokladu, že bude zachované oznámenie o copyrighte a o tom, že distribútor príjemcovi poskytuje povolenie na ďalšie šírenie, a to v rovnakej podobe, akú má toto oznámenie.

**Žilinská univerzita v Žiline**  
**Spoločnosť pre otvorené informačné technológie**

**OTVORENÝ SOFTVÉR VO VZDELÁVANÍ,  
VÝSKUME A V IT RIEŠENIACH**



**Zborník príspevkov medzinárodnej konferencie  
OSSConf 2015**

**1.–3. júla 2015  
Žilina, Slovensko**

**Recenzenti:**

Blaško, Rudolf, RNDr., PhD., Grondžák, Karol, doc., Ing., PhD., Kaukič, Michal, Mgr., CSc., Kovalík Štefan, RNDr., PhD., Kozubík, Aleš, RNDr., PhD., Majer, Tomáš, Ing., PhD., Márton Peter, doc., Ing., PhD., Peško, Štefan, doc., RNDr., PhD., Rybička Jiří, doc., Ing., Dr., Stříž, Pavel, Ing., Ph.D., Zaitseva Elena, doc., Ing., PhD.

**Editori:**

Rudolf Blaško, Aleš Kozubík

Všetky práce, uverejnené v zborníku, boli posúdené dvomi nezáujatými recenzentmi.

Práce, uverejnené v zborníku, neprešli jazykovou úpravou.

# Obsah

|   |            |
|---|------------|
| <b>Úvod</b>   | <b>5</b>   |
| <b>Jiří Rybička</b><br>Otevřený software pro řízení modelové železnice (vyžádaná přednáška) . . . . .   | 7          |
| <b>Lenka Anstead, Ľuboš Balážovič</b><br>Geospatial analysis of selected catchment parameters in Open source GIS to aid flood prevention and management . . . . . | 13         |
| <b>Rudolf Blaško</b><br>Sadzba QR Kódov . . . . .   | 21         |
| <b>Zuzana Borčinová</b><br>Použitie networkx pri riešení okružnej dopravnej úlohy . . . . .   | 29         |
| <b>Tomáš Hála</b><br>Proč jsem zkusil ConT <sub>E</sub> Xt . . . . .  | 37         |
| <b>Michal Chovanec</b><br>Required value classification using Kohonen neural network . . . . .  | 41         |
| <b>Aleš Kozubík</b><br>Mapy v L <sup>A</sup> T <sub>E</sub> X-ových dokumentoch – predstavenie balíčka <code>getmap</code> . . . . .                              | 47         |
| <b>Alena Pešková</b><br>Insight maker: simulácia údržby a obnovy strojov . . . . .  | 55         |
| <b>Vladimír Piliš, Ľuboš Balážovič</b><br>LMS Chamilo v nižšom sekundárnom vzdelávaní . . . . .   | 63         |
| <b>Jan Přichystal</b><br>Návrhář stylů v aplikaci T <sub>E</sub> XonWeb . . . . .   | 71         |
| <b>Jan Růžička</b><br>Routing v PostGIS nad daty silniční databanky . . . . .   | 77         |
| <b>Pavel Stríž</b><br>Balíček <code>xpic</code> : komutativní diagramy . . . . .  | 85         |
| <b>Peter Šarafín, Veronika Olešnaníková, Róbert Žalman</b><br>Meranie CO <sub>2</sub> s využitím platformy YROBOT . . . . .                                       | 89         |
| <b>Martin Varga, Miroslav Kvaššay</b><br>Optimization of Variables Ordering in Binary Decision Diagrams using Importance Measures . . . . .                       | 95         |
| <b>Abstrakty nerecenzovaných ukážok a prezentácií</b>   | <b>105</b> |



Vážení čitatelia,

dostáva sa vám do rúk zborník, ktorý je výstupom v poradí už siedmej samostatnej konferencie, venovanej slobodnému a otvorenému softvéru a jeho využitiu vo vzdelávaní, vede a ostatných oblastiach, ktoré si vyžadujú IT riešenia. Hoc v poradí siedma, ostáva naša konferencia stále prvou a jedinou konferenciou na Slovensku, venovanou otvoreným technológiám.

A kam sa za týchto, pre niekoho dlhých, pre niekoho krátkych sedem rokov posunul otvorený softvér? Je neprehliadnutelné, že aj na tomto poli nastal významný pokrok. Možno aj vďaka nám sa už otvorený softvér dostal do povedomia širšej užívateľskej spoločnosti. Nik sa už nepozastavuje nad tým, ak povie, že bežíte po Linuxom. Takmer nestretnete užívateľa, ktorý by nepoznal Firefox, bežne používame smartfóny a tablety s Androidom a ani záverečné práce sádzané v TEX-u už nie sú výnimočné. Žiaľ, len v oblasti verejnej a štátnej správy stále zaostávame za svetovým trendom a sporadické výskyt OSS vo verejnom živote sú stále len ojedinelými lastovičkami, ktoré jar nerobia. Ostáva nám veriť, že aj predchádzajúce ročníky nášho OSSConfu napomohli k rozšíreniu otvorených technológií a nádejať sa, že v budúcnosti napomôže aj k prelomeniu ľadov v tej poslednej menovanej oblasti.

Tento ročník konferencie prináša niekoľko tradičných sekcií. Medzi takéto „zabehnuté“ sekcie patrí „L<sup>A</sup>T<sub>E</sub>X a jeho priatelia“, ktorá má na všetkých ročníkoch svoje pevné miesto a dlhodobo patrí medzi príspevkovo najbohatšie sekcie. Za svoju životaschopnosť vďačí najmä nesmiernej aktivite jej hlavného organizátora, neúnavného Pavla Stríža.

Druhou takouto pevne zakotvenou sekciou je „Open GIS“, ktorá taktiež nechýbala ani na jednom ročníku OSSConf. Aj keď v minulom ročníku došlo k určitému poklesu prispievateľov, sme radi, že sa ku sekcii po roku vracia jej zanieteny organizátor Milo Ofúkaný. Rád vyslovujem presvedčenie, že sa mu v budúcich ročníkoch podarí vrátiť túto sekciu do pozícií, ktoré jej právom patria.

Tretou z tradičných sekcií je „OSS vo vede a vzdelávaní“. Táto sekcia je asi obsahovo najrôznorodjšia. Vďaka svojej širokospektrálnosti nemala nikdy núdzu o hodnotné príspevky, nemá však svojho jednoznačného lídra tak ako dve vyššie uvedené a obávam sa, že pri vymenovávaní jednotlivcov by som určite na niekoho zabudol. Snáď sa teda nikoho nedotkne, ak v súvislosti s touto sekciou spomeniem aspoň Miloša Šrámečka a Michala Kaukiča, ktorí stáli aj pri zrode samotnej konferencie ako celku a aktívne sa angažovali aj v oblasti vzdelávania učiteľov informatiky. V ostatných ročníkoch sa v tejto sekcii angažuje aj Martin Šechný.

Novinkou v našom programe je prvý samostatný blok, ktorý je venovaný otvorenému hardvéru. Ukazuje sa, že aj táto oblasť otvorených technológií je dostatočne životaschopnou a chcem vysloviť nádej, že sa do budúca stane rovnako pevnou súčasťou našich konferencií. Jej hlavný garant sa možno vykryštalizuje práve spomedzi jej dnešných priekopníckych účastníkov.

Na tento ročník sme pre Vás pripravili celkovo tri vyžiadané prednášky a ďalších 30 zaujímavých príspevkov. Veríme, že každý si medzi nimi nájde svoje záujmy ako aj dostatok nových podnetov a inšpirácií. rovnako radi prijmeme aj Vaše podnety a nápady o tom, čo by ste radi videli a počuli na našom ďalšom ročníku. Ako sa zvykne v takýchto prípadoch hovoriť, iniciatívy sa medze nekladú.

Záverom by som sa chcel osobitne poďakovať vedeniu Fakulty Riadenia a Informatiky ŽU zastúpenému osobou jej dekana pána doc. Ing. Emila Kršáka, PhD., za dlhoročnú podporu našej konferencie a bezodplatné poskytnutie konferenčných miestností a laboratórií v priestoroch fakulty.

Na úplný záver chcem popriať všetkým účastníkom veľa poučenia a príjemných zážitkov v komunite priaznivcov otvoreného softvéru a otvorených technológií.

Za organizačný výbor OSSConf2015

Aleš Kozubík  
predseda



## OTEVŘENÝ SOFTWARE PRO ŘÍZENÍ MODELOVÉ ŽELEZNICE (VYŽIADANÁ PREDNÁŠKA)

JIŘÍ RYBIČKA (CZ)

**Abstrakt.** Modelová železnice již v dnešní době není pouhá relativně jednoduchá hračka. Ukazuje se, že může sloužit jako velmi sofistikovaná učební pomůcka, a to jak pro studenty při seznamování s možnostmi řízení technologických procesů, tak pro výcvik dispečerů a výpravčích; vhodný železniční model lze využít jako historický doklad o technice určité doby nebo jako ukázkou procesů v nejrůznějších dalších souvislostech. S postupem miniaturizace v elektronice, zlevňováním komplikovaných součástí a snahou co nejvíce se přiblížit modelované realitě se železniční modely stávají propracovanými a relativně velmi složitými technologickými celky, jejichž ovládání je již zcela běžně prováděno počítači – stolními, přenosnými, tablety, chytrými telefony. K tomu je potřeba nejen odpovídající technické vybavení, ale také vhodné programy. Kromě různých proprietárních firemních řešení je k dispozici také obecně koncipovaný program Rocrail, jehož vlastnostmi se zabývá tento příspěvek.

**Klíčová slova.** digitální řízení, železniční model, dekodér, DCC, MTB, RocRail.

### OPEN SOURCE SOFTWARE FOR RAILWAY MODEL CONTROL (INVITED LECTURE)

**Abstract.** Model railways is no more just a relatively simple toy. It can serve as a very sophisticated teaching aid, both for students to familiarize with the possibilities of process management, and Training dispatchers and dispatchers. A suitable railway model can be used as a historical document about the technique of time or as a demonstration of processes in a variety of other contexts. With the miniaturization of electronics, cheaper complicated components and tendency as much as possible to approach modeled reality rail models are becoming sophisticated and relatively very complex technological units, whose control is already quite common performed by computer – desktop, portable, tablets, smartphones. For this control is necessary not only to appropriate technical equipment, but also the appropriate programs. In addition to various proprietary business solutions are widely available Rocrail conceived program, which features are discussed in this article.

**Keywords.** digital control, railway model, decoder, DCC, MTB, RocRail.

## Úvod

Železnice je fenomén, který svého času byl schopen ovlivnit veškeré dění. Ve 20. století byla přeprava zboží nákladu a osob po železnici tak rozšířená, že se vžilo sousloví „železnice – páteř dopravy“. Specifická zařízení, stroje a doplňky, to všechno bylo mimo jiné také inspirací pro vývoj hraček se železniční tematikou. Vliv počítačových technologií, miniaturizace elektronických součástí a

jejich zlevňování přinesl i do této oblasti zcela převratné možnosti. Relativně jednoduché modely obsahující stejnosměrný motor řízený pouze velikostí napětí v napájecích kolejnicích nebo troleji byly vystřídány digitálně řízeným strojem s mnoha funkcemi, které otevírají široké možnosti dalšího podstatného přiblížení modelované realitě a umožňují využití tam, kde dřívější technika selhávala. Objevily se proto zcela přirozeně i nové možnosti v oblasti vzdělávání – simulátory reálného provozu pro výcvik železničních zaměstnanců, ale také modely pro výuku na nejružnějších stupních a zaměřeních škol.

## Technické vybavení

Pro vytvoření železničního modelu potřebujeme vyřešit dvě oblasti – vozidla (hnací i pasivní, tedy lokomotivy i vozy) a trať se všemi komponentami (výhybky, návěstidla, výkolejky, rozpojovače, úseky kolejí). Pro současné technologie je typické, že tyto oblasti se řeší odděleně – zvláště pro řízení vozidel a zvláště pro řízení traťových komponent. Můžeme pro obojí využít stejného řešení, ale ani to není nutné.

Hnací vozidla obsahují stejně jako dříve stejnosměrný trakční motor pohánějící přes různé převody hnací kola, ale oproti dřívějšímu je tento motor vícepólový (má mnohem plynulejší chod a možnost práce při velmi nízkých otáčkách), navíc je řízen nikoliv hladinou stejnosměrného napětí, ale šířkou pulzů, což umožňuje velmi jemné ovládání a lepší chod. V hnacím vozidle je vložen dekodér, který ze střídavého napájecího napětí vytváří stejnosměrné pulzní napětí a ovládá šířku těchto pulzů.

Střídavé napájecí napětí má obdélníkový průběh a je v něm současně modulován digitální signál pro všechny dekodéry, které se nacházejí na trati. Digitální signál má obvykle formát podle normy NMRA [3], jedná se o datové pakety, jejichž součástí je digitální adresa. Paket je tedy zpracováván pouze tím dekodérem, jehož vnitřní adresa souhlasí s adresou paketu. Všechny lokomotivy na trati tedy „poslouchají“ všechny povely, ale jen ty, které jsou určeny pro ně, realizují. To umožňuje oproti minulosti zcela nezávislé chování poměrně velkého počtu hnacích vozidel na stejném úseku trati, tedy nezávislý posun ve stanicích, jízdu vlaku s příprěží, odstavení lokomotiv na manipulačních kolejích nebo v depu atd. Dekodér má navíc další funkce – je schopen ovládat světla nezávisle na rychlosti jízdy, zvuky, generování kouře, spřáhla, dveře atd.

Ani vozy už nemusí být obyčejnými pasivně taženými vozidly bez jakékoliv aktivity. Nejrozšířenější je možnost osvětlení osobních vozů, a to s různými zajímavými funkcemi – simulace „nabíhání“ zářivek při rozsvěcování, dělení vozu na oddíly se samostatným ovládním osvětlení, využití zadních koncových světel vlaku atd. Jsou k dispozici i modely, které mají digitálně ovládaná spřáhla a otevírání dveří. Pro účely detekce přítomnosti v traťovém úseku bývají vozy vybavovány speciálně upravenými nápravami, umožňujícími to, co je k dispozici

v reálné železniční dopravě – proudovým snímačem detekovat přítomnost jakéhokoliv vozu v daném traťovém úseku.

Traťové komponenty pracují na přibližně stejném principu. Mohou být ovládány signálem, který je stejný jako signál pro vozidla, ale putuje po druhé sběrnici, nebo může být použit jiný formát a typ sběrnice pro získání jiných technických výhod.

Téměř žádná modelová trať se neobejde bez výhybek. Pomineme-li možnost ručního přestavování, je k dispozici celá škála přestavníků. Od známých elektromagnetických, postavených na principu přitahování kovového jádra do prostoru cívky až po mnoho typů motorických přestavníků, které mohou věrně modelovat skutečný proces přestavení výhybky pomocí drátovodů. Stejně jako u vozidel má každý přestavník svůj dekodér s digitální adresou a datovým paketem jdoucím po sběrnici je aktivována příslušná funkce na příslušném přestavníku. Nejde jen o povel k přestavení, ale také třeba dotaz na současnou polohu či povel k rozsvícení výhybkového návěstidla.

Detekce přítomnosti vozidla v určitém místě je zcela nezbytná pro nejrůznější situace. Jednou z typických aplikací je automatický provoz přejezdového zařízení (závory, světelná signalizace). Vozidlo, které mine určité místo na trati, spustí závory nebo signalizaci na blízkém přejezdu, po přejetí celého vlaku se opět přejezd deaktivuje.

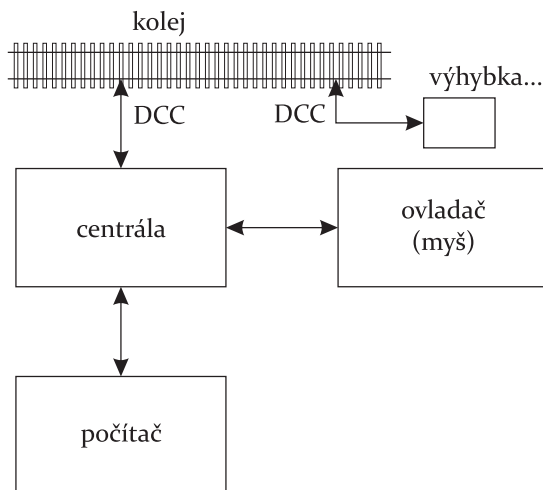
Detekce přítomnosti vozidla je nezbytná rovněž pro automatický provoz a simulaci dispečerského ovládání. To lze zařídit několika způsoby. Nejblíže realitě je již zmíněný proudový detektor snímající nenulový proud mezi kolejnicemi. Stačí tedy libovolné hnací vozidlo, osvětlený vůz nebo každá náprava ošetřená přechodovým rezistorem pro zjištění obsazenosti úseku. Výhodou tohoto řešení je souvislá detekce celého úseku a detekce libovolného jiného i částečně vodivého předmětu (položená ruka, zapomenutý šroubek). Druhou používanou technikou je optické snímání (infračervené LED pro snímání světla odraženého od podvozku vozů, počítání projetých náprav, optické snímání stínu vrženého vozidlem apod.).

Reálný provoz je těžko myslitelný bez návěstidel. Mechanická a světelná návěstidla představují opět zařízení s vlastními dekodéry, jejichž přestavování je řešeno podobně jako u výhybek, jen počet poloh může být větší. Jedno pětisvětelné návěstidlo může mít třeba deset různých stavů. Mechanická návěstidla mohou být přestavována motorickými přestavníky podobně jako výhybky.

## Řízení počítačem

Malá domácí kolejiště jsou snadno říditelná jen pomocí základního ovladače, například tzv. myši. Jde o zařízení s vestavěným ovládacím kolečkem pro nastavení rychlosti a směru jízdy a s tlačítky pro volbu vozidla a funkce dekodéru. Stejným ovladačem se adresují i digitální komponenty trati. Takto jsou koncipovány

různé „start sety“ – kolejový ovál s jednou nebo dvěma výhybkami příležitostně sestavený na stole nebo na koberci.



**Obrázek 1.** Blokové schéma digitálního řízení (příklad zjednodušené varianty řízení jednou sběrnici formátem DCC)

Pokud ovšem chceme modelovat poněkud náročnější provoz a budujeme stabilní kolejiště, velmi brzy nebude takové jednoduché ovládání stačit. V tom okamžiku se nabízí ovládání pomocí počítače, který zjednoduší a zrychlí posílání vhodných ovládacích paketů hnacím vozidlům a komponentám trati. Doplnění základní sestavy ovládacích prvků připojením počítače ukazuje obr. 1. Je zde ukázána jedna z variant, kdy celé kolejiště je připojeno pouze dvěma vodiči, po nichž putuje napájecí střídavé napětí rozvedené kolejemi, toto napětí je pulzně modulováno řídicími pakety určenými jak pro ovládání vozidel, tak i pro příslušenství trati.

Firmy nabízející základní digitální výbavu zároveň k některým (obvykle dražším) zařízením dodávají také nějaký program pro řízení tohoto hardwaru. Jako příklad můžeme zmínit aplikace pro mobilní zařízení k centrále Z21 nebo program Rocomotion firmy Roco. Alternativní řízení je relativně mnoho, zajímavou možností představuje například simulátor představený v práci Krénara [1].

## Program RocRail

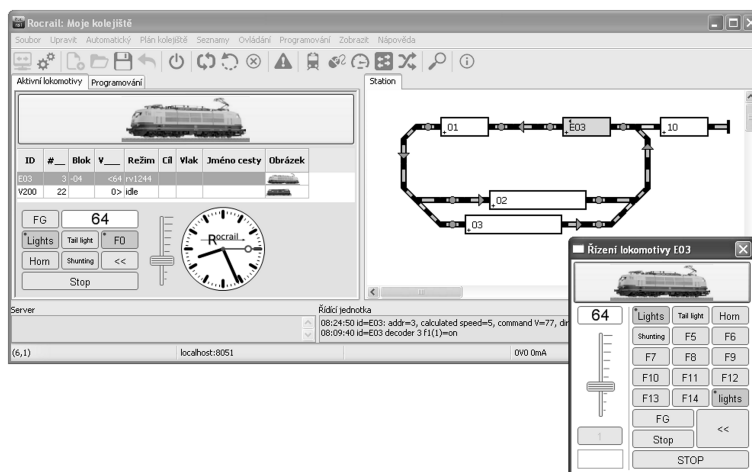
Alternativou uvedených firemních řešení programového řízení kolejového provozu, která pochází ze skupiny Open Source programů, je program Rocrail. Je použitelný na více platformách včetně linuxových [5] [4].

Popišme některé základní vlastnosti tohoto programu. Vzhledem k jeho otevřenému charakteru je velkou výhodou, že do určité míry lze přizpůsobit některé

vlastnosti potřebám uživatele, minimálně tím, že ovlivní texty zobrazované v uživatelském rozhraní. Program je k dispozici v mnoha jazykových variantách, ale ne všechny řetězce jsou ještě přeloženy.

Rocrail má připraveny ovladače pro celou škálu různých technických zařízení a typů centrál. Pokud tedy vlastníme technické řešení některé firmy, je vysoce pravděpodobné, že tento typ centrály a formát digitálního přenosu nalezneme v seznamu podporovaných zařízení.

Program se skládá ze dvou částí – serverová část (Rocrail) udržuje spojení s hardwarem a běží na pozadí, klientská část (Rocview) zobrazuje potřebné informace a komunikuje s uživatelem. Ukázkou uživatelského rozhraní lze vidět na obr. 2. Serverová a klientská část komunikuje přes TCP/IP a nemusí být instalována na jednom počítači. Klientská část může být provozována i přes webový prohlížeč, kolejiště lze tedy ovládat i přes internet.



**Obrázek 2.** Ukázka uživatelského rozhraní programu Rocview

Možnosti řízení zahrnují všechny běžné potřeby – od editace vlastního schématu kolejiště s potřebnými prvky (výhybky, návěstidla, řízené bloky trati) přes zavedení lokomotiv a sestavení vlaků až po poloautomatický provoz podle navrženého jízdního řádu. Těmito možnostmi přesahuje tento program většinu vlastností komerčně dodávaných programů, které jsou orientovány spíše na ovládání vozidel než na model provozu skutečné železnice.

## Technické řešení kolejiště Klubu modelářů železnic Brno I

Pro rozsáhlejší kolejiště je nezbytné, aby sběrnice pro ovládání vozidel a sběrnice pro digitální prvky kolejiště byly odděleny. Je technicky možné, aby každá z těchto sběrnic komunikovala jiným protokolem. V tomto smyslu byla vyvinuta

i technologie MTB pro digitální prvky tratě [2]. Protože systém MTB je lokální záležitostí Klubu železničních modelářů v Brně, není pochopitelně předdefinována podpora tohoto technického vybavení v programu Rocrail. Je proto otázkou dalšího vývoje, zda tuto podporu doplnit. Pro ovládání kolejiště totiž slouží i lokálně vyvinutý program, jehož chování je přizpůsobeno jednak technickému vybavení tratě, jednak také vzhledu a chování Jednotného obslužného pracoviště (JOP) Českých drah. Máme tak možnost simulovat reálné řízení provozu podobným způsobem, jakým pracují dispečeri na Českých drahách. V tomto ohledu bohužel program Rocrail má zcela odlišnou koncepci.

Technologie MTB a ovládací software je používán v Laboratoři řízení kolejových vozidel na Provozně ekonomické fakultě Mendelovy univerzity. Pro možnost práce studentů při vývoji ovládacích programů a rozšiřování možností technického vybavení je zcela nepochybně užitečné, pokud se s programem Rocrail seznámí a budou ho moci alternativně využít pro řešení provozních situací na trati. Cesta přizpůsobení programu Rocrail použité technologii je tedy žádoucí a tímto směrem bude také veden budoucí výzkum.

## Závěr

Volně dostupný a otevřený software má svou nezastupitelnou roli i v takových aplikacích, jako je řízení technologických procesů – tedy i řízení kolejových vozidel a technického vybavení tratě. Máme-li k dispozici takovou alternativu, je možné jednak využít jejich možností pro samotné řízení, ale i pro výuku a tvůrčí práci.

## Reference

- [1] KRÉNAR, D.: *Analýza modelu řízení železnice*, Bakalářská práce, Mendelova univerzita v Brně, 2014.
- [2] TRÁVNÍK, P.: *Systém řízení kolejiště MTB*, <http://www.kmz-brno.cz/view.php?cislocclanku=2007100007>, [cit. 20. 6. 2015].
- [3] DCC Working Group, <http://www.nmra.org/dcc-working-group>, [cit. 20. 6. 2015].
- [4] Rocrail: *Step by step tutorial*, <http://wiki.rocraail.net/doku.php?id=stepbystep-en>, [cit. 20. 6. 2015].
- [5] Rocrail: <https://en.wikipedia.org/wiki/Rocraail>, [cit. 20. 6. 2015].

## Kontaktní adresa

**doc. Ing. Jiří Rybička, Dr.**, Ústav informatiky, Provozně ekonomická fakulta, Mendelova univerzita v Brně, Zemědělská 1, 613 00 Brno, Česká republika, *Aktuální adresa*: SOIT, 010 01 Žilina, Slovenská Republika,  
*E-mailová adresa*: [rybicka@mendelu.cz](mailto:rybicka@mendelu.cz), <http://akela.mendelu.cz/~rybicka>

## GEOSPATIAL ANALYSIS OF SELECTED CATCHMENT PARAMETERS IN OPEN SOURCE GIS TO AID FLOOD PREVENTION AND MANAGEMENT

LENKA ANSTEAD (SK) AND LUBOŠ BALÁŽOVIČ (SK)

**Abstract.** The evaluation of flood risk in small catchments in Slovakia integrating open source GIS modules presents a viable tool in flood assessment that is not yet well adopted. Considering the complexity of relationships, the lack of hydrological information about small catchments and the variable character of climatic mechanisms, flood risk evaluation presents a challenging task. This paper aims to introduce a scheme for methodology of catchment analysis demonstrated on a model basin of Tajovský brook near Banská Bystrica, Slovakia. Certain modules in GRASS are shown and the pitfalls and problems dealt with at the early stages are discussed. The scheme presented will be tested and integrated into a complex flood risk evaluation project that uses hydrological records, maps and field data in open source HN models and GIS to produce inundations maps showing the depth and the extent of flood events.

**Key words and phrases.** catchment retention, hydrodynamic modeling, GRASS, flood risk, open source GIS.

## PRIESTOROVÁ ANALÝZA VYBRANÝCH PARAMETROV POVODIA V PROSTREDÍ OPEN SOURCE GIS NA PREVENCIU A MANAŽMENT POVODNÍ

**Abstrakt.** Hodnotenie povodňového ohrozenia v malých povodiach na Slovensku pomocou open source GIS predstavuje nástroj, ktorého potenciál nie je celkom využitý. Vzhľadom na zložitosť vzťahov, nedostatok hydrologických informácií o malých povodiach a variabilný charakter klimatických mechanizmov, predstavuje hodnotenie povodňového ohrozenia náročnú úlohu. Tento článok prezentuje schému metodológie analýzy povodia v modelovom území Tajovského potoka pri Banskej Bystrici. Predstavujeme vybrané moduly GRASS a diskutujeme o úskaliach a problémoch, na ktoré sme narazili v počiatkových fázach analýz. Prezentovaná schéma bude otestovaná a začlenená do projektu komplexného hodnotenia povodňového ohrozenia, ktorá integruje hydrologické údaje, mapy a dáta z terénneho mapovania v open source HN modeloch a GIS s cieľom vytvorenia inundačnej mapy znázorňujúcej hĺbku a rozsah povodne.

**Kľúčové slová.** retenčná schopnosť povodia, hydrodynamické modelovanie, slobodný GIS, GRASS, povodňové ohrozenie.

### Introduction

Flood events are becoming increasingly more frequent during the last decades, mostly caused by rapid downpour from frontal and convective weather events.

The infrastructure and property damage caused by high flows is significant, especially by those that occurred in the recent years. The intensity of these events is thought to be increasing in the future. Based on climate scenarios [3] catchments in Slovakia should be expecting a significant increase in the frequency of extreme rainfall events from short-term convective storms by up to 50% which can lead to increased flows by more than 100 % compared with historical floods. Although the average annual discharges are slightly decreasing, it is the timing and intensity of the extreme weather events that is causing floods [8].

The studied catchment of Tajovský stream was repeatedly hit by flash floods in the recent years. It is therefore important to analyse what catchment parameters influence the flood events the most in order to mitigate the effects of future events. Based on the European and national legislation (EU Directive 2007/60/ES), Slovakia should address the problems concerning flood risk and flood hazard maps. Flood hazard maps should show the areas of flood inundation, flood risk maps should report on the economic and social effects of the inundation.

GIS represent an essential tool in research, evaluation and management of flood risk. Geospatial approach allows for a complex analysis of multiple parameters and in the conjunction with modeling tools, it creates a powerful tool in predicting and timing the flood events. Open source GIS can be especially useful, thanks to its availability, ability to independently test and develop new modelling modules, open code and developers community support which lacks with commercially available software. Most OS GIS is available under the GPL (GNU Public license), handles a variety of data and formats, can be used for spatial analysis and modeling and runs on a variety of platforms including Linux or Macintosh.

Various projects were carried out to integrate and standardise methods and to provide modelling support tools for flood risk estimation across Europe (such as Harmonique, FLOODsite, EXCIMAP), [4]. In Slovakia, this effort was concentrated on large catchments and only on selected urban reaches that were in the risk of flooding. Commercially available packages were used in the analysis such as hydrodynamic models MIKE 11, MIKE 21, MIKE FLOOD (DHI Water & Environment), FLO-2D (FLO-2D Inc., USA) but also in open source HEC-RAS (US Army Corps of Engineers). These were visualised and further processed in ArcGIS or in AutoCAD [4] into flood maps that are now available online <http://mpomprsr.svp.sk>. Small catchments are, however, not included and are planned to be processed in the future.

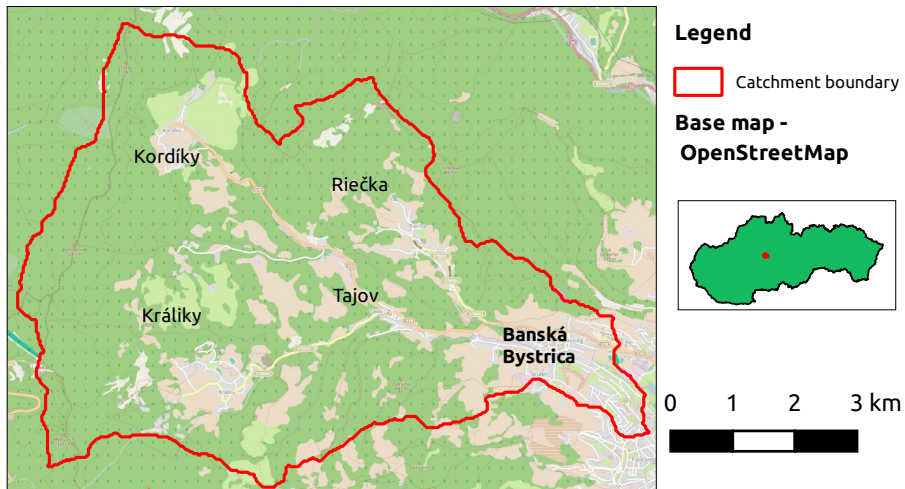
With regards to small catchments (under 250 km<sup>2</sup>), retention functions and flood hydrographs have been studied on a basis of hierarchical cluster analysis, or by a method of regional frequency analysis for the estimation of N-year flows [6] or by integrating digital network of small watersheds and GIS [7]. Some work based on GIS approach for modelling flood hydrograph in small ungauged catchments was undertaken by Danko [1], but a complex GIS study integrating modelling



approach that would result in flood hazard estimation is missing. The aim of this paper is therefore to provide different datasets which will be suitable for flood wave modelling and in the production of flood inundation map.

## 1. The study area

The studied catchment of Tajovský brook is located near Banská Bystrica in Central Slovakia (Fig. 1). It covers area of around 40 km<sup>2</sup>. Tajovský brook is the right tributary of the River Hron, the second longest river in Slovakia and it is 11.8 km long from its source to the confluence. It originates on the southeast slopes of Skalka in Kremnické hills in altitude of 1,150 m a.s.l. The river network reflects the lithological and tectonic conditions of the area such as the strength of rocks and the position of tectonic faults. The density of riverine network is 1.74 km/km<sup>2</sup> and consists of 7 main tributaries.



**Figure 1.** The model catchment of Tajovský stream (Base map: OpenStreetMap, 2015)

The climate of the catchment is classified as temperate to cold [3]. Maximum precipitation occurs in June and July due to convective storms, minimum occurs in January. Summer precipitation intensity can reach up to 200 litres per second/hectare. Rainfall is increasing with increasing altitude up to around 1300 mm/year. The snow cover lasts between 60 and 100 days. There are three rain gages located in the catchment and one flow gage at the down end of the catchment. The highest culmination flow of  $30.7 \text{ m}^3 \cdot \text{s}^{-1}$  occurred in August 2002, when it reached the second stage of flood warning. Table 1 shows peak discharges since 2000.

**Table 1.** Values of peak annual discharges in the profile Tajovský brook since 2000 (SHMÚ, 2011)

| Year             | 2000  | 2001   | 2002   | 2003  | 2004 | 2005  | 2006  | 2007   | 2008  | 2009    | 2010   | 2011   | 2012    |
|------------------|-------|--------|--------|-------|------|-------|-------|--------|-------|---------|--------|--------|---------|
| Date             | 9. 3. | 17. 9. | 12. 8. | 3. 1. | 6. 8 | 19. 3 | 1. 4. | 19. 1. | 1. 3. | 25. 12. | 27. 9. | 20. 7. | 27. 10. |
| Q <sub>max</sub> | 17.04 | 11.76  | 30.69  | 4.72  | 7.77 | 10.88 | 8.35  | 7.12   | 11.44 | 9.91    | 12.97  | 11.56  | 8.08    |

## 2. Methodological scheme

When starting to evaluate the retention capacity of catchment to rainfall, the methodology presented by [1] for small subalpine and alpine basins was used. On the input, we have, initially, integrated these layers in GRASS: (1) elevation field from topographic map SVM50, (2) aerial photographs with 1m resolution and (3) the map of soil-ecological units (BPEJ) or soil types. In further steps, other inputs will be used for generating specific maps for the analysis (Fig. 2).

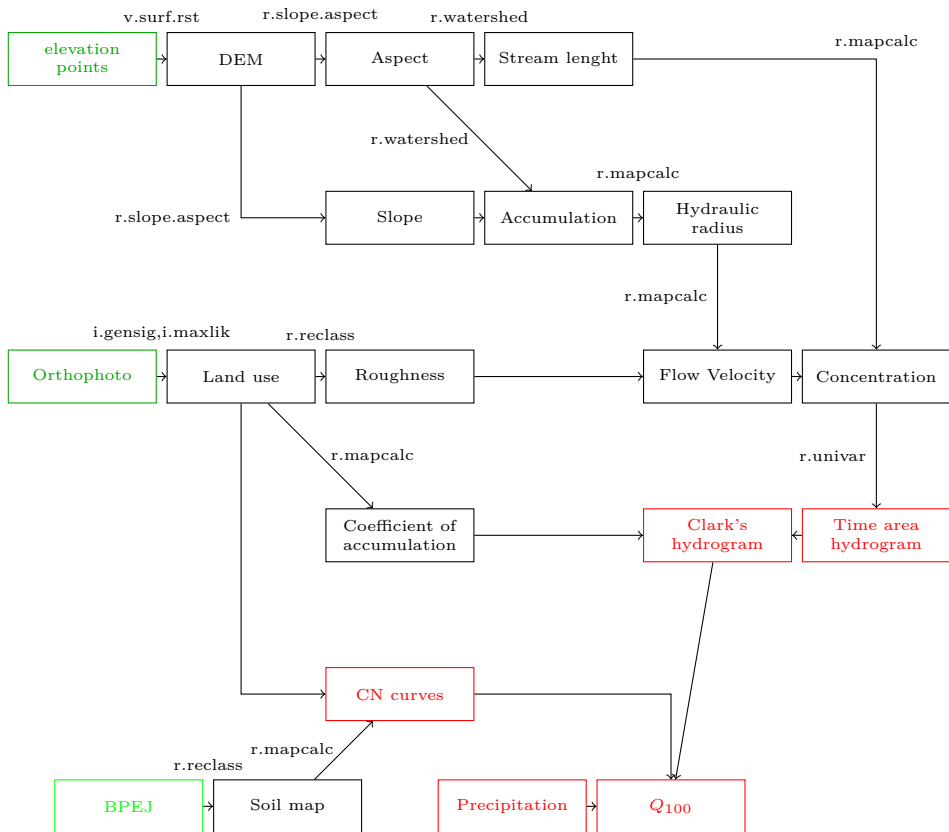
Number of selected maps were created from the source maps: DEM, slope map, flow direction map, flow accumulation, map of hydraulic radius, map of land cover and Manning's roughness map.

### 2.1. DEM

From the elevation field of the catchment, we have created a coarse DEM using interpolation method regularised spline with function tension (module `v.surf.rst`) where tension was set to 10 (Fig. 3). The catchment boundary was generated through `r.watershed` module, setting minimum size to 30000 cells. The river network generated in the output DEM agreed well with the one on the SVM50. Best results were seen in the upper parts of catchment, bigger discrepancies were identified on the floodplain (up to 200 m). Further analysis is needed to quantify the accuracy and compare various sources of riverine network. Where the floodplain was wider, the differences between the original map and the generated river network were bigger (up to 200 m). According to the hypsographic classification by Mazúr (1980), from `r.reclass` of the DEM, low highlands (300–750 m a.s.l.) cover 67.23 per cent and medium highlands (750–1500 m a.s.l.) cover 32.77 per cent of the catchment. The highest elevation was identified as 1281 m a. s. l. and the lowest 340 m a. s. l.

### 2.2. Slope map

The module `r.slope.aspect` generated maps of slope and aspect. The module uses  $3 \times 3$  neighbouring cells to calculate slope (in degrees) and consequently module `r.cost` is used to calculate the cumulative cost of traversing the slope map starting from the stream channel. The value of the slope for a cell is the cost of traversing the cell. Cells with low slope, near the stream channel or in flood plain, will have lower total cost of traversing, when compared to the cells farther

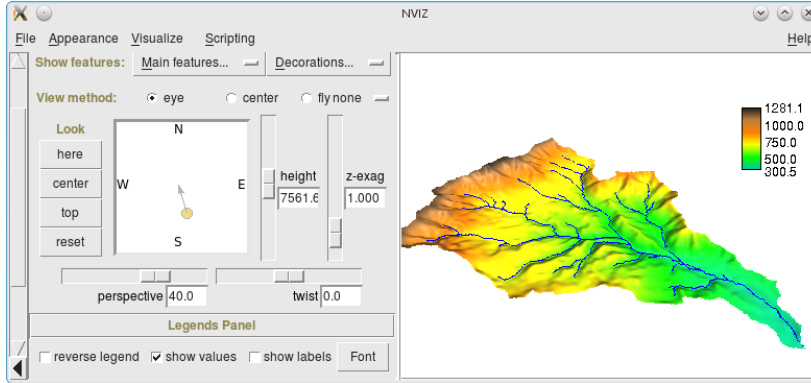


**Figure 2.** GRASS implementation of flood wave model according to Danko (2011)

away from the stream. This cost will increase as steeper cells are included, as will be the case with cells that are away from the floodplain. Among the many such paths possible, water flows to the path of least resistance or lowest slope cost. The module `r.cost` generated a map containing the least cumulative cost of traversing each cell from the starting cell, which in this case, are the points on the stream.

### 2.3. Maps of flow direction and accumulation

From digital elevation model (DEM) flow direction map was created using the `r.watershed` module. The output was the map of the direction of the steepest descent of each cell of the grid to neighboring low-lying cell. Based on the analysis of flow directions, with the function on flow accumulation, runoff accumulation map can then be created together with the runoff micro network paths of concentrated surface runoff, depending on the size of the contributing area. Same



**Figure 3.** Catchment 3D visualisation in NVIZ

module was used to produce a flow accumulation map that shows the number of cells that drain each cell. It indicated places of flow accumulation, drainage, location to streams and watersheds.

#### 2.4. Map of hydraulic radius

The map was derived from the accumulation map using constants for  $Q_{100}$  according to the equation [5]:

$$R = a(M_a * 0.01)^b$$

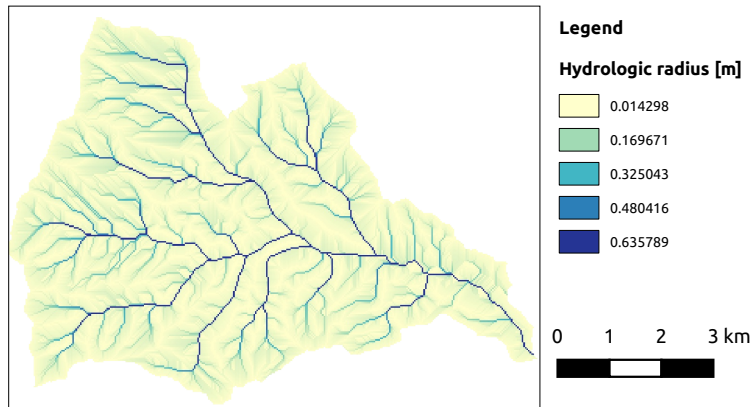
Where  $R$  is hydraulic radius [m],  $a$  is a constant that for  $Q_{100}$  equals to 0.18,  $b$  is a constant that for  $Q_{100}$  equals to 0.55 and  $M_a$  is the value of accumulation identified from the accumulation map. Hydraulic radius is a hydrological parameter described as the ratio of cross-sectional area of the channel to the wetted perimeter. Module `r.mapcalc` was used to create the output map (4).

#### 2.5. Landuse map

An 2004 orthophoto map (1m res.) was imported using `r.in.gdal` module. Training map of sample polygons was created and imported using `v.in.ogr`. The layer was converted from vector to raster using `v.to.rast.attr`. Supervised classification was used to identify land use categories (e.g. forest, grassland, field, urbanised area) using `i.gensig` for generating signatures and `i.maxlik` for maximum likelihood. From the initial coarse analysis, 55% of the area was grassland and fields, 48% is forest and 7% are urbanised areas. Land use classified into some further categories provides a basis for the Manning's roughness map, where each category is given an estimated number or a specific code of a HN model using module `r.map.calc`. This map is essential for flood modelling and can be derived using various inputs (LiDAR, orthophoto, CORINE, OSM), depending on

the data availability and the accuracy required [2]. Corine Landcover was not used in this analysis because it may be not fine enough for small catchments. However, in further stages of the project, roughness maps and the effect on flood curve generated from models using Corine will be compared with outputs generated from high resolution aerial and Sentinel images.

To complete the procedure of flood wave estimation, further data and map resources need to be acquired that were not available in the time of writing this paper. As a final output, the flood inundation map will be created. This will integrate both the data from the catchment and the estimated hydrological values (as shown on Fig. 2) as well as field geodetic and hydrological data in a real time hydrological model.



**Figure 4.** Hydraulic radius

### 3. Conclusions

Evaluation of flood hazard in small catchments in order to produce data necessary for flood prevention and management, such as the estimation of certain hydrologic or spatial parameters, is a very complex issue. This paper introduced just a small piece of the methodology that can be implemented as a part of the process using open source GIS tools. Further input data, secondary maps and integration with HN models is necessary to establish the most important hydrological characteristics and produce flood inundation maps of a given area. A key challenge would be to reduce the number of resources, steps and resolution into a level that would give reliable results for flood hazard estimation. This is important especially for developing and implementing an approach that would be widely applicable to small catchments in Slovakia as such methodology does not exist.

However, the specific conditions of each catchment may prove it as a challenging task. As a final conclusion, open source GIS demonstrated an abundance of modules and functions that can be used in hydrological analysis of small catchments and the next step will be to integrate the GIS tools with open source hydrological models to prove its versatility for this application.

## References

- [1] DANKO, M., HLAVČOVÁ, K., KOHNOVÁ, S., SZOLGAY, J.: (2011) Modelling of design flood wave in small catchments without observations using GIS. *Acta Hydrologica Slovaca* (12) 2, 340-348.
- [2] DORN, H., VETTER, M., HÖFLE, B.: *Derivation for Flood Simulations: A Comparison of Orthophotos, LiDAR and Crowdsourced, Remote Sensing*, (2014) 6, 1739–1759, DOI:10.3390/rs6021739, ISSN 2072-4292.
- [3] LAPIN, M.: *Changes in Meteorological Conditions of Flood Risk in the Changing Climate*, Životné prostredie, Vol. 37, No. 4 (2003).
- [4] LUKÁČ M. et al: *Hydronumerical modelling and EU flood directive*, Vodohospodársky spravodajca, (2010) 9–10, 26–29, ISSN 0322-886X.
- [5] MOLNAR, P., RAMIREZ, J. A.: *Energy dissipation theories and optimal channel characteristics of river networks*, Water Resources, (1998) 34 (7), 1809–1818.
- [6] SOLÍN, L.: *Floods – estimation of their magnitude in small catchments in Slovakia using regional frequency analysis*, Geografický časopis, (2005) 57, 287–307.
- [7] SOLÍN, L.: *Analysis of flood occurrence in Slovakia between 1996 and 2006*, Journal of Hydrology and Hydromechanics, (2008) 56 (2), 95–115.
- [8] SZOLGAY, J. et al.: *Analysis of flood risk in the upper Hron catchment*, Záverečná správa VTP 27–34 Výskum vplyvu antropogénnych faktorov na vodné systémy, časť etapy E 04.01, 2002, VÚVH, Bratislava.
- [9] VLČKO, F.: *Analysis and trends of atmospheric precipitation in Tajovský catchment*, Bakalárska práca, 2012, Univerzita Mateja Bela v Banskej Bystrici, Fakulta prírodných vied, 61 s.

## Contact addresses

**Mgr. Lenka Anstead, PhD.**, Department of Geography and Geology, Faculty of Natural Sciences, Matej Bel University, Tajovského 40, 974 01 Banská Bystrica, Slovakia,  
*E-mail address:* lenka.anstead@umb.sk, <http://www.fpv.umb.sk/lanstead/>

**Mgr. Ľuboš Balážovič, PhD.**, Department of Geography and Geology, Faculty of Natural Sciences, Matej Bel University, Tajovského 40, 974 01 Banská Bystrica, Slovakia,  
*E-mail address:* lubos.balazovic@umb.sk, <http://www.fpv.umb.sk/lbalazovic/>

## SADZBA QR KÓDOV

RUDOLF BLAŠKO (SK)

**Abstrakt.** Dojrozmerné rovinné QR kódy, na rozdiel od jednorozmerných čiarových kódov, majú oveľa širšie využitie medzi bežnými ľuďmi. Vyvinula ich japonská spoločnosť Denso-Wave v roku 1994 a patria medzi otvorené štandardy. Ich špecifikácia je štandardom ISO/IEC 18004. Vytvoriť vlastný QR kód môžeme viacerými spôsobmi. Najjednoduchšie je použiť niektorý z voľne dostupných generátorov QR kódov na internete. Druhá, oveľa zaujímavejšia možnosť, je vygenerovať si QR kód samostatne pomocou  $\TeX$ -u.

**Kľúčové slová.** QR kód, sadzba,  $\LaTeX$ , PSTricks, balíčky `pst-barcode`, `auto-pst-pdf`, `qrcode`.

### QR CODES TYPESETTING

**Abstract.** In contrast to the one-dimensional bar codes the two-dimensional plane QR codes have a much wider use among common people. They have been developed by the Japanese company Denso-Wave in the 1994 and they belong among the open standards. Their specification is ISO/IEC 18004. Own QR code can be created by several ways. The simplest is to use one of the free QR code generators available on the Internet. The second, more interesting possibility is to generate a QR code oneself using  $\TeX$ .

**Keywords.** QR Code, typesetting,  $\LaTeX$ , PSTricks, packages `pst-barcode`, `auto-pst-pdf`, `qrcode`.

## Úvod

QR kódy (z anglického „Quick Response Code“ – kódy rýchlej odozvy, reakcie) sú kódy, ktoré na rozdiel od klasických čiarových EAN kódov, sú dvojrozmerné a dokážu zašifrovať oveľa väčšie množstvo údajov (nemusia to byť iba url adresy alebo informácie o tovare alebo platbe). QR kódy môžu obsahovať ľubovoľnú textovú informáciu – url, e-mailovú adresu, telefónne číslo a jeho následné vytočenie v mobilnom telefóne, zamilovaný pozdrav pre svoju lásku atď. QR kód patrí medzi otvorené štandardy a vyvinula ho japonská spoločnosť Denso-Wave v roku 1994 (spoločnosť zverejnila špecifikáciu a neuplatňuje patentovú ochranu). Špecifikácia QR kódov je štandardom ISO/IEC 18004 z roku 2000.

Najčastejšie sa QR kódy používajú:

- webová adresa alebo e-mailová adresa,
- kontaktné informácie (formát vCard alebo MECARD),
- odoslanie preddefinovanej SMS alebo MMS,

- spustenie telefónneho hovoru s definovaným telefónnym číslom,
- uloženie poznámky do mobilného kalendára,
- načítanie GPS súradníc,
- ľubovoľný zakódovaný text alebo odkaz.

QR kód sa skladá z bielych a čiernych štvorcových modulov zložených do štvorcovej matice. Veľkosť matice je od  $21 \times 21$  (verzia/veľkosť 1) do  $177 \times 177$  bodov (verzia/veľkosť 40) s nárastom o 4 moduly v každom rozmere. QR kód môže obsahovať písmená, čísla alebo japonské znaky kandži a je schopný pojať rôzne množstvo informácií podľa obsahu (čísllice – 7089 znakov, písmená a čísllice – 4296 znakov, 8-bitové dáta – 2953 znakov, kandži – 1817). Farby nemusia byť iba biela a čierna, pri rozpoznávaní QR kódu nie je podstatné jeho otočenie a nevyžaduje sa ani veľký farebný kontrast. Každá verzia QR kódu má maximálnu prenosovú kapacitu v závislosti od množstva a typu dát a úrovne opravy chýb. QR kód má v sebe obsiahnutú korekciu chýb, to znamená schopnosť obnoviť dáta v prípade, že kód je znečistený alebo poškodený. Schopnosť QR kódu pri oprave chýb má štyri úrovne s rôznou korekciou chýb (L  $\approx$  7%, M  $\approx$  15%, Q  $\approx$  25%, H  $\approx$  30%). Je zrejmé, že tu platí nepriama úmernosť — čím vyššia korekcia chýb, tým menej dát v QR kóde [6].

QR kódy môžu byť statické (ich obsah je nemenný) alebo dynamické (odkazujú sa na špeciálne servery, kde sa ukladajú informácie o kódovaných údajoch a výsledné odkazy je možné podľa potreby prispôsobovať, t. j. nie je potrebné meniť QR kód). Cieľom príspevku nie je špecifikácia a vnútorná štruktúra QR kódu, ale iba jeho praktické vytvorenie pomocou dostupných vybraných prostriedkov — free online generátorov, resp. vybraných balíčkov L<sup>A</sup>T<sub>E</sub>X-u.

## 1. Online nástroje na tvorbu QR kódov

Na webe sa vyskytuje veľké množstvo generátorov na tvorbu QR kódov, ktoré sú bezplatné. Spomenieme niektoré z nich (QR kódy url adresy webovej stránky autora <http://frcatel.fri.uniza.sk/~beerb> vygenerované pomocou nasledujúcich online QR generátorov sú na obr. 1):

- **QR Code Generator** (<http://goqr.me/>) – jednoduchý interaktívny generátor, QR kód sa generuje priamo pri písaní textu, je ohraničený 300 znakmi a výsledný obrázok je možné stiahnuť vo formátoch .svg, .eps, .png a .jpg. Dovoľuje generovať statické a aj dynamické QR kódy a poskytuje možnosť nastaviť veľkosť kódu od 1 do 10, korekciu chýb a farby popredia a pozadia. V menu sú voľby pre generovanie QR kódu pre webovú adresu, text, Vcard, SMS, telefónne číslo, GPS, kontakt (napr. do mobilného telefónu) a e-mail.
- **QR code for free** (<http://www.qr-code-generator.com/>) – dovoľuje generovať statické a aj dynamické QR kódy pre prakticky neobmedzené



množstvo textu, výsledný obrázok je možné stiahnuť vo formátoch `.svg`, `.eps`, `.jpg` a `.pdf`, ale je potrebná free registrácia (e-mail a heslo). V menu sú voľby pre generovanie QR kódu pre webovú adresu, text, Vcard, SMS.



Obr. 1. QR kódy adresy `http://frcatel.fri.uniza.sk/~beerb` vygenerované pomocou uvedených online generátorov (po rade zľava doprava)

- **QR Code Generator** (<https://www.the-qr-code-generator.com/>) – interaktívny generátor, ktorý generuje statický QR kód priamo pri písaní textu, výsledný obrázok je možné stiahnuť vo formátoch `.svg`, `.eps` a `.png`. V menu sú voľby pre generovanie QR kódu pre webovú adresu, text, SMS, telefónne číslo a kontakt.
- **QR Stuff.com** (<http://www.qrstuff.com/> – generátor, ktorý generuje statické aj dynamické QR kódy bez registrovania vo formáte `.png`. V menu sú voľby pre generovanie QR kódu pre rôzne špecifikácie (webová adresa, text, Vcard, SMS, YouTube, Bitcoin, Twitter ap.).
- **Morovia – Free online QR Code Maker** (<http://www.morovia.com/free-online-barcode-generator/qr-code-maker.php>) – pomerne jednoduchý generátor, ktorý dovoľuje nastavenie modulov v pixeloch (1–4), veľkosť (auto, resp. 1–40) a korekciu chýb (obr. 2). Výsledný obrázok je možné stiahnuť pomocou prehliadača (pravé tlačítko myši, voľba „uložiť obrázok ako“) vo formáte `.png`. Okrem QR kódov je možné generovať aj rôzne EAN kódy (<http://www.morovia.com/free-online-barcode-generator/>).

Existuje samozrejme oveľa viac generátorov QR kódov. Nie je problém po niekoľkých sekundách strávených na internete nájsť ďalšie vyhovujúce online generátory. Druhou možnosťou ako generovať QR kódy je investovať trochu času do vzdelania a vytvoriť ich pomocou  $\text{\LaTeX}$ -u.

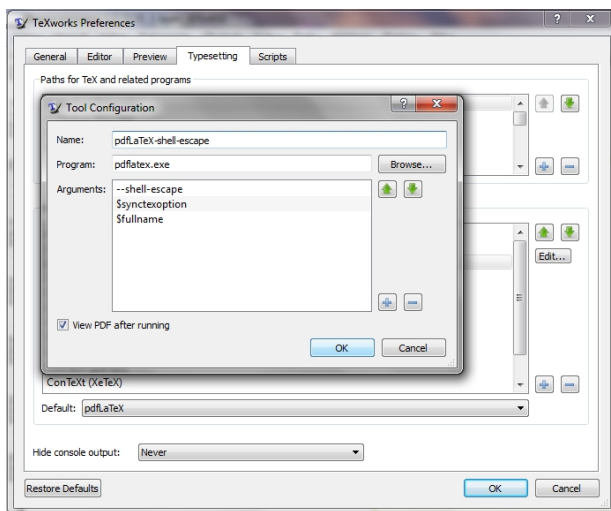


Obr. 2. QR kódy adresy `http://frcatel.fri.uniza.sk/~beerb` s rôznymi úrovňami schopnosti opravy chýb (úrovne L, M, Q, H zľava doprava) vygenerované pomocou generátora Morovia – Free online QR Code Maker

## 2. Sadzba QR kódov pomocou balíčka `pst-barcode`

Pri sadzbe QR kódov pomocou balíčka `pst-barcode` nemôžeme použiť priamo preklad pomocou `pdflatex`, pretože tento balíček patrí do skupiny makier, ktoré zastrešuje postskriptový balíček `PSTricks`. V takýchto prípadoch pri generovaní súborov typu `.pdf` musíme voliť zložitejšiu cestu `.dvi` → `.eps` → `.pdf` na príkazovej úrovni (ako v Linuxe, tak aj vo Windows). To znamená, že musíme najprv náš súbor (napr. `subor.tex`) preložiť pomocou  $\text{\LaTeX}$ -u `latex subor.tex`. Dostaneme súbor `subor.dvi`, ktorý pomocou programu `dvips` prekonvertujeme príkazom `dvips subor.dvi` na súbor `subor.ps`. Zostáva ešte prekonvertovať súbor `subor.ps` príkazom `ps2pdf subor.ps` na súbor `subor.pdf`.

Tento proces sa dá automatizovať pomocou balíčka `auto-pst-pdf`. Po použití tohto balíčka môžeme súbor `subor.tex` preložiť priamo do formátu `.pdf` pomocou príkazu `pdflatex --shell-escape subor.tex`. Ak používame niektorý z editorov  $\text{\TeX}$ -u, je potrebné zabezpečiť preklad s parametrom `--shell-escape`. Nastavenie editora  $\text{\TeX}$ works je znázornené na obr. 3. Nastavenie je identické v Linuxe aj vo Windows a nachádza sa v menu `Edit` — `Preferences...` — `Typesetting`.



**Obr. 3.** Nastavenie editora  $\text{\TeX}$ works s parametrom `--shell-escape` pre balíčky `PSTricks` pre preklad do súboru `.pdf`

Pomocou balíčka `pst-barcode` môžeme sádzať rôzne jednorozmerné čiarové kódy (napr. EAN-13, EAN-8, EAN-5, EAN-2, ISBN, Pharmacode) alebo dvojrozmerné kódy (napr. QR kódy, Data matrix, Aztec kódy, 2D Maxi kódy). Tejto problematike sa venoval P. Stríž na OSSConf 2011 [4]. Všetky tieto kódy sa sádzajú pomocou príkazu

`\psbarcode[voľby]{kódovaný text}{PS voľby}{typ barkódu}`,

ktorý **používa syntax PSTricks** a je umiestnený v prostredí `pspicture` alebo v boxe `\makebox{}`. Výsledný obrázok je sádzaný do boxu s nulovými rozmermi.

Parameter `voľby` je nepovinný a môže obsahovať horizontálne alebo vertikálne posunutie `transx`, `transy` (implicitne `transx=0`, `transy=0` v palcoch), horizontálne alebo vertikálne škálovanie `scalex`, `scaley` (implicitne `scalex=1`, `scaley=1`), otočenie o uhol `rotate` (implicitne `rotate=0`), ale tiež rôzne nastavenia, ktoré používa PSTricks, napr. `linecolor=red` definuje farbu čiar na červeno.

Postskriptové PS voľby dovoľujú určiť rôzne nastavenia (viď [5], str. 5) ako sú napr. veľkosť kódu (implicitne `height=1` v palcoch), veľkosť a pozícia textu, vkladanie nového textu (relevantné pri čiarových kódoch).



```
\documentclass[12pt]{article}
\usepackage{pst-barcode}
\usepackage{auto-pst-pdf}

\begin{document}
  \begin{pspicture}(0.7in,0.7in)
    \psbarcode{http://frcatel.fri.uniza.sk/~beerb}{}{qrcode}
  \end{pspicture}
\end{document}
```

Obr. 4. QR kód adresy `http://frcatel.fri.uniza.sk/~beerb` vygenerovaný pomocou balíčka `pst-barcode`

Posledný parameter `typ barkódu` určuje aký kód sa bude vlastne sádzať (napr. `ean13`, `ean8`, `upca`, `upce`, `isbn`, `code128`, resp. parameter `qrcode` pre QR kódy).

Pri sádzaní QR kódu môžeme medzi PS voľby pridať nastavenia pre korekciu chýb `eclevel=L` (ďalšie možnosti sú `M`, `Q`, resp. `H`), pre formát symbolov `format=micro` alebo `format=full`. Ďalej je možné nastaviť verziu `version=1` až `version=40` pre formát `full` (korekcia chýb je implicitne nastavená na `M`), resp. `version=M1` až `version=M4` pre formát `micro` (korekcia chýb je implicitne nastavená na `L`). Ešte môžeme nastaviť typ znakov napr. `encoding=alphanumeric`, `encoding=numeric`, `encoding=byte`, `encoding=kanji`.



Obr. 5. QR kódy adresy `http://frcatel.fri.uniza.sk/~beerb` vygenerované pomocou balíčka `pst-barcode` s rôznymi parametrami

Na obr. 5 sú QR kódy adresy `http://frcatel.fri.uniza.sk/~beerb` vygenerované postupne zľava doprava príkazmi:

```
\psbarcode[linecolor=red]{http://fr ... sk/~beerb}{}{qrcode},
\psbarcode{http://frcatel.fri.uniza.sk/~beerb}{version=6}{qrcode},
\psbarcode{http://fr ... sk/~beerb}{version=6 eclevel=Q}{qrcode},
\psbarcode[linecolor=red,rotate=45]{http://fr ... sk/~beerb}{}{qrcode}.
```

### 3. Sadzba QR kódov pomocou balíčka qrcode

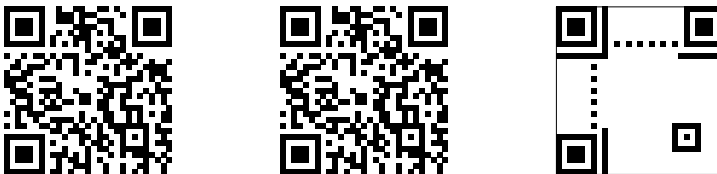
Balíček sa načítava príkazom `\usepackage[voľby balíčka]{qrcode}` v preambule a obsahuje jediný príkaz `\qrcode`, ktorý má nasledujúcu syntax:

```
\qrcode[voľby]{kódovaný text}.
```

Balíček spolupracuje s balíčkom `hyperref` a ak je tento balíček načítaný, príkaz `\qrcode` automaticky aktivuje hyperlinkové prepojenie na danú internetovú lokalitu, resp. e-mailovú adresu (obr. 6 vľavo).

Parameter `voľby balíčka` pri načítavaní balíčka je nepovinný a môže nadobúdať hodnoty:

- nolink** : Ak je načítaný balíček `hyperref`, voľba `nolink` deaktivuje hyperlinkové prepojenie (obr. 6 v strede). Ak uvedený balíček nie je načítaný, táto voľba nemá žiadny efekt.
- draft** : Táto voľba spôsobí, že sa negeneruje obrázok QR-kódu (pri väčších textoch môže generovanie QR kódu dlhšie trvať), ale sa zobrazí iba stručný náčrt QR kódu (obr. 6 vpravo).
- final** : Pri tejto voľbe sa QR kód generuje iba vtedy, ak nie je ešte zaznamenaný v súbore `.aux`. Ak je súbor `.aux` odstránený, pri ďalšom preklade sa QR kód opäť generuje.
- forget** : Pri tejto voľbe sa QR kód generuje vždy.



**Obr. 6.** QR kódy adresy `http://frcatel.fri.uniza.sk/~beerb` vygenerované pomocou balíčka `qrcode` bez parametrov (vľavo), s parametrom `nolink` (v strede) a s parametrom `draft` (vpravo)

Najpoužívanejšie hodnoty nepovinného parametra `voľby` príkazu `\qrcode` sú:

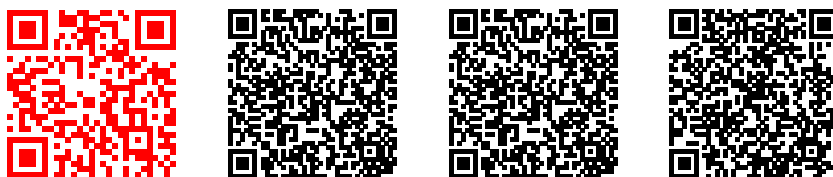
**height** : Nastavuje šírku a výšku QR kódu (napr. `height=.5\linewidth`), implicitné nastavenie je `height=2cm`.

**level** : Nastavuje korekciu chýb (L, M, Q alebo H — implicitne `level=M`).

**version** : Nastavuje verziu/veľkosť od 1 do 40, implicitne je nastavená voľba `version=0`, ktorá znamená výber najmenej novej hodnoty.

**link resp. nolinek** : Ak je načítaný balíček `hyperref` tieto voľby aktivujú, resp. deaktivujú hyperlinkové prepojenie.

Namiesto príkazu `\qrcode[nolinek]{kódovaný text}` môžeme použiť jeho skrátený hviezdičkový ekvivalent `\qrcode*[voľby]{kódovaný text}`.



Obr. 7. QR kódy adresy `http://frcatel.fri.uniza.sk/~beerb` vygenerované pomocou balíčka `qrcode` s rôznymi parametrami

Na obr. 7 sú QR kódy adresy `http://frcatel.fri.uniza.sk/~beerb` vygenerované postupne zľava doprava príkazmi:

```
{\color{red}\qrcode{http://frcatel.fri.uniza.sk/~beerb}},
\qrcode[version=6]{http://frcatel.fri.uniza.sk/~beerb},
\qrcode[version=6,level=Q]{http://frcatel.fri.uniza.sk/~beerb},
\qrcode*[version=6,level=Q]{http://frcatel.fri.uniza.rsk/~beerb}.
```

#### 4. Záver

QR kódy, t. j. kódy rýchlej reakcie, sú v poslednom čase veľmi populárne a asi aj budú. Ich výhodou je, že sa môžu použiť v tlačenej forme (letáky, plagáty, obaly kníh ap.) a ich obrázok môžeme zosnímať nejakým mobilným zariadením (napr. mobilný telefón, tablet) a automaticky sa presmerovať na danú lokalitu alebo vytočiť dotyčného adresáta. Rovnakú úlohu spĺňajú aj v elektronickej forme. Buď ich zosnímame iným elektronickým médiom alebo jednoducho klikneme na daný odkaz. Ich veľkému rozšíreniu určite napomohla aj krajina ich vzniku, kde sú elektronické hračky rôznych druhov veľmi populárne.

**Podakovanie.** Tento príspevok bol podporený grantom slovenskej kultúrno-edukačnej agentúry KEGA č. 011ŽU-4/2014 „Experimentálna matematika – zviditeľnenie neviditeľného“.

## Literatúra

- [1] HENDRICKSON, A.: *The qrcode package: Quick Response code generation in L<sup>A</sup>T<sub>E</sub>X*, January 8, 2015, [cit. 20.6.2015], <http://mirrors.nic.cz/tex-archive/macros/latex/contrib/qrcode/qrcode.pdf>, resp. <http://ftp.cvut.cz/tex-archive/macros/latex/contrib/qrcode/qrcode.pdf>.
- [2] KOZUBÍK, A.: *Naučím vás kresliť alebo predstavenie balíčka TiKZ*, Otvorený softvér vo vzdelávaní, výskume a IT riešeniach, zborník medzinárodnej konferencie OSSConf 2012, Žilina, 2.–4. júla 2012, str. 91–96, ISBN 978-80-970457-2-2, <http://ossconf.soit.sk/images/zborniky/zbornik2012.pdf>.
- [3] KUBEŠ R.: *QR kódy*, časopis CHIP 06/2013, [cit. 20.6.2015], <http://www.chip.cz/casopis-chip/earchiv/vydani/rocnik-2013/chip-06-2013/qr-kody/>.
- [4] STRÍŽ, P.: *Typografie a čárové kódy*, Otvorený softvér vo vzdelávaní, výskume a IT riešeniach, zborník medzinárodnej konferencie OSSConf 2011, Žilina, 6.–9. júla 2011, str. 199–207, ISBN 978-80-970457-1-5, <http://ossconf.soit.sk/images/zborniky/zbornik2011.pdf>.
- [5] VOSS, H., BURTON, T.: *Pst-barcode, A PStricks package for drawing bar codes; v.0.14*, June 17, 2015, [cit. 20.6.2015] <http://mirrors.nic.cz/tex-archive/graphics/pstricks/contrib/pst-barcode/doc/pst-barcode-doc.pdf>, resp. <http://ftp.cvut.cz/tex-archive/graphics/pstricks/contrib/pst-barcode/doc/pst-barcode-doc.pdf>.
- [6] QRcode.sk, [cit. 20.6.2015], <http://www.qrcode.sk/qrkody-qrcode.php>.

## Kontaktná adresa

**RNDr. Rudolf Blaško, PhD.**, Katedra matematických metód a operačnej analýzy, Fakulta riadenia a informatiky, Žilinská univerzita, Univerzitná 8215/1, 010 26 Žilina, Slovenská Republika,

*E-mailová adresa:* [beerb@frcatel.fri.uniza.sk](mailto:beerb@frcatel.fri.uniza.sk), <http://frcatel.fri.uniza.sk/~beerb/>

## POUŽITIE NETWORKX PRI RIEŠENÍ OKRUŽNEJ DOPRAVNEJ ÚLOHY

ZUZANA BORČINOVÁ (SK)

**Abstrakt.** V tomto článku chceme predstaviť použitie NetworkX – knižnice programovacieho jazyka Python, ktorá umožňuje nenáročnú a pohodlnú implementáciu algoritmov teórie grafov. Ako ilustračný príklad sme pre svoj zámer zvolili riešenie kapacitnej dopravnej okružnej úlohy použitím základného genetického algoritmu.

**Kľúčové slová.** Python, NetworkX, kapacitná dopravná okružná úloha, genetický algoritmus.

### USING NETWORKX FOR SOLVING THE VEHICLE ROUTING PROBLEM

**Abstract.** In this paper, we introduce NetworkX – Python language software package, which enables effortless and comfortable implementation of graph algorithms. As an illustrative example we have chosen to solve the Vehicle Routing Problem using a basic genetic algorithm.

**Keywords.** Python, NetworkX, Capacitated Vehicle Routing Problem, genetic algorithm.

## Úvod

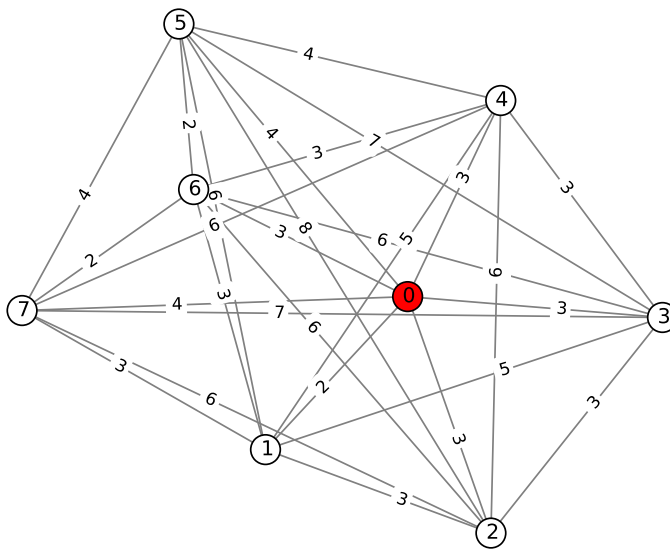
Okružná dopravná úloha (*Vehicle Routing Problem*, VRP) je jedným z kombinatorických optimalizačných problémov v oblasti prepravy, distribúcie a logistiky. Úloha VRP spočíva v hľadaní množiny ciest pre skupinu vozidiel (*vehicles*), sústredených v jednom alebo viacerých skladiskách (*depots*), ktoré zásobujú určený počet miest alebo zákazníkov (*customers*) geograficky rozložených na mape, pričom celkové náklady majú byť minimálne. Najvšeobecnejšou verziou tejto úlohy je kapacitná dopravná okružná úloha (*Capacitated Vehicle Routing Problem*, CVRP).

## 1. Opis problému

CVRP je definovaný na ohodnotenom neorientovanom grafe  $G = (V, H, c)$  s množinou vrcholov  $V = \{0, 1, 2, \dots, n\}$  (obr. 1). Vrchol 0 zodpovedá skladu, ktorý disponuje množinou  $A$  identických vozidiel s rovnakou kapacitou  $q$ . Vrcholy 1 až  $n$  reprezentujú zákazníkov (množina  $K$ ), požiadavky ktorých sú dané hodnotami  $d_i$ . Cestovné náklady medzi vrcholmi  $i$  a  $j$  sú dané ako  $c_{ij}$ , čo je cena hrany  $(i, j)$ .

Cieľom je nájsť množinu okružných jászd (trás) s minimálnou celkovou cenou, ktorá spĺňa nasledujúce vlastnosti [3]:

1. Každá trasa začína a končí v sklade.
2. Každý zákazník je navštívený práve raz.
3. Pri žiadnej trase nie je prekročená kapacita vozidla.



Obr. 1. Graf G s cenami hrán

## 2. Implementácia v Pythone

NetworkX [4] je modul programovacieho jazyka Python [5] určený na prácu so sieťami. Pomocou neho môžeme načítavať a ukladať siete v štandardných, ale aj v neštandardných dátových formátoch, generovať množstvo náhodných i klasických typov sietí, analyzovať štruktúru sietí, vytvárať modely sietí, navrhovať nové algoritmy, vykresľovať siete a podobne.

Pri implementácii CVRP budú vstupnými údajmi kapacita vozidiel  $q$  a zoznam  $Z$  usporiadaných trojíc  $(x, y, d)$ , kde  $x, y$  sú súradnice vrcholu (umiestnenie skladu alebo zákazníka) a  $d$  je požiadavka zákazníka, napríklad pre graf z obr. 1:

$$q = 50, \quad Z = [(5.4, 7.68, 0), (3.66, 5.48, 10), (6.42, 4.28, 10), (8.52, 7.38, 20), \\ (6.54, 10.5, 30), (2.6, 11.6, 10), (2.78, 9.22, 20), (0.68, 7.48, 20)].$$

Nech cestovné náklady  $c_{ij}$  sú úmerné vzdialenosti medzi vrcholmi  $i$  a  $j$ , teda v Pythone ich určíme takto:



```

from math import sqrt

def c(i,j):
    dx = i[0]-j[0]
    dy = i[1]-j[1]
    d = sqrt(dx*dx + dy*dy)
    return int(d)

```

Množinu  $H$  ohodnotených hrán môžeme reprezentovať zoznamom usporiadaných trojíc  $(i, j, c_{ij})$ , čo v Pythone zapíšeme nasledovne:

```

v = len(Z)
H = [(i,j,c(Z[i],Z[j])) for i in range(v) for j in range(i+1,v)]

```

Teraz už jednoducho vytvoríme hranovo ohodnotený graf  $G$ :

```

import networkx as nx

G = nx.Graph()
G.add_weighted_edges_from(H)

Požiadavky zákazníkov budú atribútmi vrcholov:

for i in range(v):
    G.node[i]['poziadavka']=Z[i][2]

```

Na vykreslenie grafu (viď obr. 1) použijeme knižnicu Matplotlib a metódu `draw_networkx()`, pričom zadáme jej parametre tak, aby sa vrcholy zobrazili v súradnicovej sústave (parameter `pos`) a aby sklad bol farebne odlíšený od zákazníkov (parameter `node_color`). Ak chceme zobrazit aj ceny hrán, najprv použijeme metódu `draw_networkx_edge_labels()`:

```

import matplotlib.pyplot as plt

sur = {i:Z[i][:2] for i in range(v)}
ceny = {(i,j):G.edge[i][j]['weight'] for (i,j) in G.edges()}
nx.draw_networkx_edge_labels(G,pos=sur,edge_labels=ceny)
nx.draw_networkx(G,pos=sur,node_color=['red']+(v-1)*['white'])

```

### 3. Genetický algoritmus

CVRP je NP-ťažký problém, a preto sa na jeho riešenie vyvíjajú heuristické a meta-heuristické metódy. Jednou z nich je genetický algoritmus (GA), ktorého princíp je inšpirovaný evolúciou a genetikou. GA pracuje s populáciou jedincov. Veľkosť populácie je daná počtom jedincov v populácii. Mierou kvality jedinca je hodnota *fitness*. Prvá populácia je vygenerovaná náhodne a následne sa obnovuje výberom, krížením, mutovaním a nahradením jedincov. Tieto kroky sa opakujú s cieľom zlepšovania jedincov v populácii. Cyklický proces obnovy populácie sa skončí po splnení vopred stanovených podmienok (napr. po dosiahnutí

maximálneho počtu iterácií bez zlepšenia a pod.). V ideálnom prípade by výsledná populácia mala obsahovať jedinca, ktorého kvalita je globálnym optimom funkcie fitness [1].

#### Genetický algoritmus zapísaný v pseudokóde:

```

1:   Inicializácia populácie Pop
2:   repeat
3:       výber dvoch rodičov R1 a R2 z Pop
4:       križenie R1 x R2 --> D1, D2
5:       for každé D z {D1, D2}:
6:           mutácia D (s malou pravdepodobnosťou)
7:           nahradenie jedného jedinca B z Pop potomkom D
8:       endfor
9:   until kritérium ukončenia je splnené
10:  riešením je najlepší jedinec v populácii Pop

```

## 4. Riešenie CVRP pomocou GA

Najprv stručne objasníme, ako budeme reprezentovať jedincov v prípade CVRP a ako fungujú operácie výberu rodičov, križenia, mutácie a nahradenia jedincov. V závere sa zameriame na podrobnejšie vysvetlenie procedúry `split` [2], ktorú použijeme na dekodovanie riešenia.

### 4.1. Jedinec a jeho fitness

Jedincom budeme rozumieť jedno potenciálne riešenie CVRP. Na reprezentáciu riešenia použijeme reťazec  $S$  obsahujúci zákazníkov zo všetkých okružných jász, ale bez oddeľovačov jednotlivých trás. Riešenie  $S$  je teda permutáciou zákazníkov  $S = (S_1, S_2, \dots, S_n)$ , kde  $S_i \in \{1, 2, \dots, n\}$  a  $S_i \neq S_j$ , ak  $i \neq j$  [3].

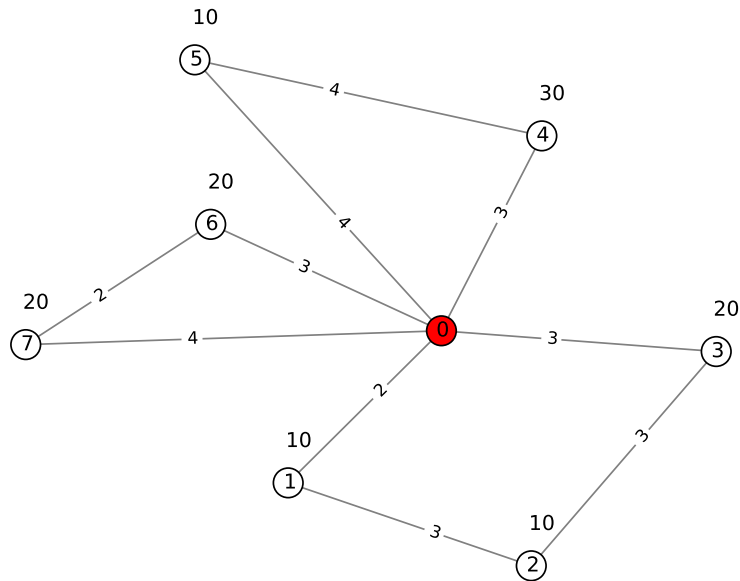
Hodnota  $f(S)$  fitness riešenia  $S$  je rovná cene riešenia, to znamená celkovým cestovným nákladom (súčet cestovných nákladov zodpovedajúcich hranám patriacim riešeniu), napr. cena riešenia  $S = (1, 2, 3, 6, 7, 4, 5)$  na obr. 2 je

$$f(S) = c_{01} + c_{12} + c_{23} + c_{30} + c_{06} + c_{67} + c_{70} + c_{04} + c_{45} + c_{50} = 31.$$

Úlohou CVRP je minimalizovať celkové cestovné náklady, a preto kvalita jedinca bude nepriamo úmerná jeho fitness, t.j. čím je hodnota fitness menšia, tým je jedinec lepší.

### 4.2. Výber rodičov

Na výber rodičov použijeme binárny turnaj. Najprv z aktuálnej populácie náhodne vyberieme dvoch jedincov a lepšieho z nich označíme ako rodiča  $R_1$ . Rovnako postupujeme pri výbere rodiča  $R_2$ .



Obr. 2. Jedno riešenie VRP,  $S = (1, 2, 3, 6, 7, 4, 5)$

### 4.3. Kríženie

Kríženie rodičov simuluje reprodukciu. Výsledkom sú noví jedinci (potomkovia)  $D_1$  a  $D_2$ , ktorí kombinujú vlastnosti rodičov. Pri jednoduchom jednobodovom krížení operátor kríženia náhodne rozdelí oboch rodičov na dve časti. Potomok  $D_1$  vznikne skopírovaním ľavej časti rodiča  $R_1$  a doplnením permutácie tak, aby sa zachovalo poradie prvkov z rodiča  $R_2$ . Pre vytvorenie potomka  $D_2$  sa úlohy rodičov vymenia, napr.

$$\begin{array}{ll} \mathbf{R}_1: 3\ 2\ 4\ 5 \mid 1\ 7\ 6, & \mathbf{D}_1: 3\ 2\ 4\ 5 \mid 1\ 6\ 7, \\ \mathbf{R}_2: 1\ 2\ 6\ 4 \mid 7\ 5\ 3, & \mathbf{D}_2: 1\ 2\ 6\ 4 \mid 3\ 5\ 7. \end{array}$$

### 4.4. Mutácia

Mutácia sa používa na náhodné malé zmeny potomkov, ktoré môžu priniesť nové charakteristiky do populácie. Operátor mutácie náhodne vyberie dva prvky z jedinca a vymení ich. K mutácii dochádza s malou pravdepodobnosťou, napr.

$$\begin{array}{ll} \mathbf{S}: 3\ 2\ \underline{4}\ 5\ 1\ \underline{6}\ 7, & \mathbf{S}': 3\ 2\ \underline{6}\ 5\ 1\ \underline{4}\ 7, \\ \text{pôvodný jedinec} & \text{zmutovaný jedinec} \end{array}$$

#### 4.5. Nahradenie

Jeden cyklus obnovenia populácie (iterácia) sa zavŕši nahradením najhoršieho jedinca v populácii potomkom  $D_1$ . Potom sa znovu vyhledá najhorší jedinec a nahradí sa potomkom  $D_2$ .

#### 4.6. Dekódovanie riešenia

Pri výpočte fitness je potrebné riešenie  $S$  optimálne rozdeliť na jednotlivé kapacitne prípustné trasy (dekódovať). Na tento účel použijeme procedúru `split` [2]. Táto procedúra vytvorí pomocný hranovo ohodnotený digraf  $D = (X, F, w)$ . Množina vrcholov  $X = \{0, 1, 2, \dots, n\}$  a množina orientovaných hrán  $F$  obsahuje tie hrany  $(i, j)$ ,  $i, j \in X$ ,  $i < j$ , pre ktoré platí, že trasa obsahujúca zákazníkov  $S_{i+1}$  až  $S_j$  je kapacitne prípustná, t. j.

$$\sum_{k=i+1}^j d_{S_k} \leq q.$$

Najkratšia orientovaná cesta z vrchola 0 do vrchola  $n$  v digrafe  $D$  udáva miesta, v ktorých bude riešenie  $S$  optimálne rozdelené na prípustné trasy a jej dĺžka zodpovedá cene riešenia, čiže fitness  $f(S)$ .

Pri programovaní si uvedomíme si, že:

- Požiadavky zákazníkov sme zadali do grafu  $G$  ako atribúty vrcholov, teda súčet požiadaviek na trase  $S_{i+1}$  až  $S_j$  vypočítame:

```
def dopyt(G,S,i,j):
    s = sum([G.node[S[k]]['poziadavka'] for k in range(i+1,j+1)])
    return s
```

- Ohodnotenie  $w_{ij}$  orientovanej hrany  $(i, j)$  je rovné cestovným nákladom trasy  $(0, S_{i+1}, S_{i+2}, \dots, S_j, 0)$ , teda:

```
def w(G,S,i,j):
    T = [0]+S[i+1:j+1]+[0]
    naklady = sum([G.edge[T[y]][T[y+1]]['weight']
                  for y in range(len(T)-1)])
    return naklady
```

- Množinu  $F$  ohodnotených orientovaných hrán budeme reprezentovať zoznamom trojíc  $(i, j, w_{ij})$ :

```
C = [0]+S[:]
p = len(C)
F = [(i,j,w(G,C,i,j)) for i in range(p-1)
     for j in range(i+1,p) if dopyt(G,C,i,j)<= q]
```

- Pomocný digraf  $D$  s množinou  $F$  ohodnotených orientovaných hrán vytvoríme:

```
D = nx.DiGraph()
D.add_weighted_edges_from(F)
```

- Najkratšiu orientovanú cestu z vrchola 0 do vrchola  $n$  v digrafe  $D$  nájdeme:

```
n = D.number_of_nodes()-1
cesta = nx.shortest_path(D,0,n,'weight')
```

- Dĺžku nájdenej najkratšej cesty zistíme:

```
fitness = nx.shortest_path_length(D,0,n,'weight')
```

Napríklad pre riešenie  $S = (6, 7, 1, 2, 3, 4, 5)$  bude cesta  $[0, 3, 5, 7]$ , a preto dekodované riešenie  $R = (0, 6, 7, 1, 0, 2, 3, 0, 4, 5, 0)$  s cenou 30 (obr. 3).

```
R = [0]+sum([S[cesta[i]:cesta[i+1]]+[0]
            for i in range(len(cesta)-1)])
```

- Ak chceme pre väčšiu názornosť vykresliť dekodované riešenie  $R$  (napr. ako na obr. 3), stačí v metóde `draw_networkx()` zadať zoznam hrán, ktoré sa majú zobrazit (parameter `edgelist`). Pre zobrazenie požiadaviek pri vrcholoch použijeme metódu `draw_networkx_labels()`, pričom určíme súradnice, kde sa majú požiadavky zobrazit (parameter `pos`) a tiež hodnoty požiadaviek (parameter `labels`):

```
hrany = [(R[i],R[i+1]) for i in range(len(R)-1)]
ceny_hran={i,j):G.edge[i][j]['weight'] for (i,j) in hrany}
nx.draw_networkx_edge_labels(G,pos=sur,edge_labels=ceny_hran)
poziadavky = {i:G.node[i]['poziadavka'] for i in G.nodes()}
labsur = {i:(Z[i][0]+0.1,Z[i][1]+0.6) for i in range(v)}
nx.draw_networkx_labels(G,pos=labsur,labels=poziadavky)
nx.draw_networkx(G,pos=sur,node_color=['red']+(v-1)*['white'],
                 edgelist=hrany)
```

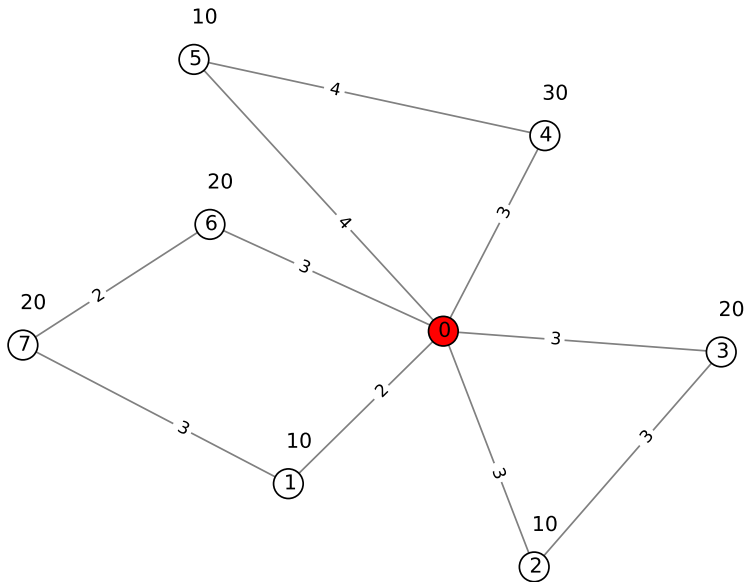
## 5. Záver

Programovací jazyk Python spolu s modulom NetworkX poskytuje možnosť jednoduchej a flexibilnej reprezentácie grafov, ako aj zrozumiteľný a prehľadný zápis grafových algoritmov. A čo je rovnako dôležité, Python je voľne šíriteľný a užívateľsky priateľský. Na prezentovanie jeho knižnice NetworkX sme zvolili riešenie CVRP pomocou genetického algoritmu pre jeho pomerne ľahkú implementáciu nenáročnú na riešiteľove hlboké znalosti matematických základov daného problému.

**Podakovanie.** Príspevok vznikol s prispením grantu KEGA 011-4/2014ŽU „Experimentálna matematika – zviditeľnenie neviditeľného“ podporeného Slovenskou kultúrno-edukačnou grantovou agentúrou.

## Literatúra

- [1] PELIKAN, M.: Evolučné algoritmy, Umelá inteligencia a kognitívna veda I., 2009, 335–353, ISBN 9788022730808.



**Obr. 3.** Riešenie  $S = (6, 7, 1, 2, 3, 4, 5)$  optimálne rozdelené pre  $q = 50$

- [2] PRINS, C.: *A simple and effective evolutionary algorithm for the vehicle routing problem*, Comput. Oper. Res., 31(12), 2004, 1985–2002.
- [3] SÖRENSEN. K. , SEVAUX, M.: *A practical approach for robust and flexible vehicle routing using metaheuristics and Monte Carlo sampling*, Journal of mathematical modelling and algorithms., 2009, pp. 387–407, ISSN 1570-1166-8.
- [4] *NetworkX documentation*, <http://networkx.github.io/documentation/latest/index.html>.
- [5] *Python 3.4.3 documentation*, <http://docs.python.org/3/>.

## Kontaktná adresa

**R.NDr. Zuzana Borčinová**, Katedra matematických metód a operačnej analýzy, Fakulta riadenia a informatiky, Žilinská univerzita, Univerzitná 8215/1, 010 26 Žilina, Slovenská republika,  
*E-mailová adresa:* [zuzana.borcinova@fri.uniza.sk](mailto:zuzana.borcinova@fri.uniza.sk)

## PROČ JSEM ZKUSIL CONTEX<sub>T</sub>

TOMÁŠ HÁLA (CZ)

**Abstrakt.** Příspěvek představuje tři příklady, kterými se liší přístup dvou typografických nadstaveb, ConT<sub>E</sub>X<sub>t</sub> a L<sup>A</sup>T<sub>E</sub>X<sub>u</sub>, obou jako následníků základního T<sub>E</sub>X<sub>u</sub>. Příklady se týkají konceptu názvů příkazů, sazby na řádkový rejstřík a vztahu integrace a balíčků.

**Klíčová slova.** ConT<sub>E</sub>X<sub>t</sub>, L<sup>A</sup>T<sub>E</sub>X<sub>u</sub>, srovnání, sazba.

### WHY DID I TRY CONTEX<sub>T</sub>

**Abstract.** Three examples dealing with the concept of macro names, grid typesetting and relation between integration and package use are shown in this contribution. They demonstrate different access of two typographic T<sub>E</sub>X-based superstructures, ConT<sub>E</sub>X<sub>t</sub> and L<sup>A</sup>T<sub>E</sub>X.

**Keywords.** ConT<sub>E</sub>X<sub>t</sub>, L<sup>A</sup>T<sub>E</sub>X, comparison, typesetting.

## Úvod

ConT<sub>E</sub>X<sub>t</sub> je software určený pro sazbu dokumentů s požadavkem vysoké kvality výstupního produktu, podobně jako L<sup>A</sup>T<sub>E</sub>X či samotný T<sub>E</sub>X. Z důvodu určité nespokojenosti s koncepcí vývoje i chováním nadstavby L<sup>A</sup>T<sub>E</sub>X si v roce 1990 vytvořil Hans Hagen pro potřeby své firmy Pragma Advanced Document Engineering nadstavbu vlastní a pojmenoval ji ConT<sub>E</sub>X<sub>t</sub>. Veřejnosti nabídl svoje řešení k volnému užití pod licencí GNU GPL. Za zmínku také stojí, že do dnešního dne se stará o další vývoj systému.

Protože ConT<sub>E</sub>X<sub>t</sub> je následníkem Knuthova T<sub>E</sub>X<sub>u</sub>, pracuje na stejném principu. Zdrojový text uživatel pořizuje libovolným prostředkem, vždy však jako textový soubor doplněný o formátovací instrukce vyjádřené značkovacím jazykem. Výsledný dokument ve formátu PDF pak vzniká překladem.

ConT<sub>E</sub>X<sub>t</sub> podporuje vše, co je potřeba pro profesionální práci [4]: barvy, konfigurovatelná pozadí, matematickou sazbu, chemickou sazbu, sazbu nelatinkových jazyků, hypertextové odkazy, prezentace, křížové odkazy, práci bibliografickými citacemi, rejstříky, vyřazení na archy atd. Integrované jsou i MetaPostová podpora nazývaná MetaFun a jazyk Lua.

Není asi možno v krátkém příspěvku představit všechny vlastnosti ConT<sub>E</sub>X<sub>t</sub>u, které činí tento prostředek zajímavou alternativou k všeobecně rozšířenějšímu L<sup>A</sup>T<sub>E</sub>X<sub>u</sub>. Proto zde uvedeme jen několik příkladů, které ilustrují patrné rozdíly v přístupu k vývoji i koncepci obou nadstaveb.

## Jednotný přístup k objektům sazby a jejich konfiguraci

U moderního programového vybavení se očekává i určitá intuitivnost. Proto se autor ConTeXtu rozhodl pro jednotný přístup k jednotlivým objektům sazby i k jejich konfiguraci – všechny objekty lze konfigurovat příkazem začínajícím prefixem `setup`, např. `\setuplayout`, `\setuppapersize`.

Jedná-li se o objekt, který se v L<sup>A</sup>T<sub>E</sub>Xu označuje slovem prostředí, pak k příkazům začínajícím prefixem `setup` přistupuje ještě dvojice příkazů s prefixy `start` a `stop`, které dané „prostředí“ ohraničují, např.: `\setupitemize/\startitemize/\stopitemize`, `\setuptable/\starttable/\stoptable`. Příkazy mající prefixy `start` a `stop` také, podobně jako v L<sup>A</sup>T<sub>E</sub>Xu, otevírají a uzavírají skupiny.

Vlastní konfigurace se pak provádí zadáním jednoslovných názvů atributů, případně uspořádaných dvojic ve tvaru *názevatributu=hodnota*, např.:

```
\setuplayout[horoffset=-1cc]
\setupitemize[broad,packed]
\setuphead[section][style=slanted,color=red]
```

Kolik parametrů má který příkaz a které atributy je možno použít, uvádí například [7]. Jedná se sice o dokumentaci staršího data, která v některých aspektech zastarala, ale i tak je dostatečně dobrá pro začátečnickovy první kroky.

## Řádkový rejstřík

Schopnost systému sázet na řádkový rejstřík je zcela základní a nezastupitelný prvek každého profesionálního systému. Tato idea však není v T<sub>E</sub>Xu podchycena – řádkový rejstřík může být dodržen bez problémů jen tehdy, jedná-li se o hladký text. Jsou-li přítomny další objekty (nadpisy, obrázky, tabulky atd.), jejich výška musí být celým násobkem řádkové rozteče. V ostatních případech se automaticky řádkový rejstřík nevytvoří a je potřeba buď ručně vložit vertikální mezery, nebo použít nějaké propracovanější řešení. Přehled významných možností jak dosáhnout řádkového rejstříku v plainT<sub>E</sub>Xu či v L<sup>A</sup>T<sub>E</sub>Xu byl prezentován dříve [6].

V ConTeXtu je problém sazby na řádkový rejstřík vyřešen přímočaře a z uživatelského hlediska velmi jednoduše – stačí nastavit jediný parametr:

```
\setuplayout[grid=yes]
```

## Balíčky vs. integrované prostředky

L<sup>A</sup>T<sub>E</sub>X je znám množstvím balíčků, které si uživatel může stáhnout a připojit k dokumentu, pokud řeší záležitost mimo funkcionality základní verze L<sup>A</sup>T<sub>E</sub>Xu. Pro běžného uživatele však tato cesta není nejlepší, jednak pro množství balíčků, jednak pro možné kolize mezi nimi.



Jako příklad uveďme sazbu tabulek. Řada balíčků řeší tutéž věc, například pro práci s tzv. dlouhými tabulkami je k dispozici sedm balíčků, formátováním sloupců s desetinnými čísly se zabývá šest balíčků, tři balíčky podporují rozšíření možností práce s barvou a pak se v archivu CTAN nachází dalších 28 balíčků pro jiné situace [5].

Pokud se týká kolizí, bylo porovnáno 11 balíčků pro sazbu tabulek, které byly zkoumány ve vzájemné interakci [8]. Z 55 možných kombinací v sedmi případech balíčky nespolečně pracovaly a v dalších čtyřech případech z principiálních důvodů ani nemohly. Jedná se tedy o jedenáct případů, což je jedna pětina(!) z celkového množství možných kombinací.

ConTeXt také umožňuje připojovat externí moduly a případně i vytvářet vlastní. Na rozdíl od L<sup>A</sup>T<sub>E</sub>Xu se však existující moduly [3] nepřekrývají. Kolize je velmi málo pravděpodobná i proto, že moduly implementují méně obvyklé záležitosti.

**Tabulka 1.** Srovnání balíčků L<sup>A</sup>T<sub>E</sub>Xu potřebných pro třídu `oss-conf` s možným řešením v ConTeXtu

| oss-conf.cls pro L <sup>A</sup> T <sub>E</sub> X<br><code>\usepackage{...}</code>         | Možné řešení v ConTeXtu |  |
|---|-------------------------|--|
|   | dostupnost              | příkaz(y), poznámka  |
| *volba <code>twoside</code>   | implicitně              | <code>\setuplayout[doublesided]</code>   |
| <code>babel</code> s volbami dle jazyků   | implicitně              |  |
| <code>[T1]{fontenc}</code><br><code>[utf8]{inputenc}</code>                               | implicitně              | v současnosti je UTF-8 nativní   |
| <code>amsmath</code><br><code>amscd</code> , <code>amssymb</code> , <code>amsfonts</code> | externí balíček         | AMSL [2]   |
| <code>graphicx</code>   | implicitně              | <code>\externalfigure</code>   |
| <code>color</code>  | implicitně              | <code>\setupcolors</code> , <code>\startcolor</code>   |
| <code>colortbl</code>   | implicitně              | tzv. natural tables  |
| <code>multicol</code>   | implicitně              | <code>\setupcolumns</code> , <code>\column</code>  |
| <code>comment</code>  | implicitně              | bloky [7, kap. 37.3]   |
| <code>\fancyhdr</code>  | implicitně              | <code>\setupheader</code> , <code>\setupheadertexts</code><br><code>\setupfooter</code> , <code>\setupfootertexts</code> |
| <code>hyperref</code>   | implicitně              | interakce [7, kap. 30]   |
| <code>eurosym</code>  | implicitně              | <code>\texteuro</code>   |
| <code>listings</code>   | implicitně              | <code>\setuptyping</code> , <code>\starttyping</code>  |
| <code>lmodern</code>  | implicitně              |  |

Pro zajímavost můžeme srovnat balíčky připojované, případně předpokládané pro použití v rámci třídy `oss-conf` [1] pro tuto konferenci, s řešením v ConTeXtu (tabulka 1). Pro úplnost dodejme, že třída `oss-conf.cls` je připravena pro překlad zdrojových textů pdfL<sup>A</sup>T<sub>E</sub>X.

Balíček `cite`, který je rovněž přítomen ve stylu `oss-conf`, nesrovnáváme s ohledem na všeobecnou složitost problematiky sazby bibliografických citací.

## Závěr

Zbývá odpovědět na otázku uvedenou v názvu článku.

Tři představené příklady, na kterých čtenář si může učinit úsudek o koncepci nadstavby ConTeXt, která – stejně jako L<sup>A</sup>T<sub>E</sub>X – je určena k tvorbě profesionálně vyhlížejících dokumentů, ukazují, jak lze uživateli jeho typografický život velmi zjednodušit, ať se již jedná o logické názvy příkazů v rámci skupiny příkazů, či o přímočaré nastavení sazby na řádkový rejstřík. Dále integrace většiny služeb do jednoho celku zbavuje uživatele nutnosti shánět, připojovat a kombinovat balíčky.

Autora tohoto článku popsany koncepční přístup zaujal natolik, že si ConTeXt vyzkoušel a totéž doporučuje i ostatním. Hlavní motivací však byla právě existence uživatelsky jednoduchého a funkčního řešení sazby na řádkový rejstřík.

## Reference

- [1] BLAŠKO, R.: soubor `oss-conf.cls`, 2012-04-04, v1.00, *Šablóna pre príspevky v slovenskom alebo českom jazyku*, <http://ossconf.soit.sk/images/ossconf2015-sk.rar>, <http://ossconf.soit.sk/index.php/o-konferencii/pre-autorov>, [cit. 15. 6. 2015].
- [2] ConTeXt Garden. *AMS- $\LaTeX$  compatibility*, <http://modules.contextgarden.net/ams1>, [cit. 15. 6. 2015].
- [3] ConTeXt Garden: *Modules and Packages for ConTeXt*, <http://modules.contextgarden.net>, [cit. 15. 6. 2015].
- [4] ConTeXt Garden: *What is ConTeXt*, [http://wiki.contextgarden.net/What\\_is\\_ConTeXt](http://wiki.contextgarden.net/What_is_ConTeXt), [cit. 15. 6. 2015].
- [5] GRAHAM, W.: *The  $\TeX$  Catalogue Online*, <http://ftp.cstug.cz/pub/tex/CTAN/help/Catalogue/bytopic.html#tables>, [cit. 15. 6. 2015].
- [6] HÁLA, T.:  *$\LaTeX$ , nebo ConTeXt? První zkušenosti se sazbou ConTeXtem*, Zpravodaj CSTUG, roč. 23, č. 1, 2013, s. 57–64, ISSN 1211-6661.
- [7] OTTEN, T., HAGEN, H.: *Exkurze do ConTeXtu*, Zpravodaj CSTUG, roč. 16, č. 2–4, 2006, s. 57–224, ISSN 1211-6661.
- [8] TALANDOVÁ, P.: *Možnosti tabulkové sazby*, Zpravodaj CSTUG, roč. 18, č. 3, 2008, s. 151–160, ISSN 1211-6661.

## Kontaktní adresa

**RNDr. Tomáš Hála, Ph.D.**, Mendelova univerzita, Provozně ekonomická fakulta,  
ústav informatiky, Zemědělská 1, 613 00 Brno, Česká republika,  
*E-mailová adresa: thala@mendelu.cz*

## REQUIRED VALUE CLASSIFICATION USING KOHONEN NEURAL NETWORK

MICHAL CHOVANEC (SK)

**Abstract.** In this paper we describe situations classification using Kohonen neural network. Results are demonstrated on line following robot, where different curves are classified. Each classified situation has corresponding required value output, which is used by low level control process. Network weights estimates are processed in real time.

**Key words and phrases.** kohonen neural network, learning controll, iterative learning, PID controll, mobile robot.

### Klasifikácia žiadanej hodnoty Kohonenovou neurónovou sieťou

**Abstrakt.** V článku je popísaná klasifikácia situácií Kohonenovou neurónovou sieťou. Výsledky sú demonštrované na robotovi, sledujúom čiaru, kde sú klasifikované rôzne typy zákrut. Každá situácia má zodpovedajúci výstup, ktorý je vstupom do nižšej úrovne riadenia.

**Keywords.** kohonenová neurónová sieť, inteligentné učenie, iteratívne učenie, PID riadenie, mobilný robot.

## Introduction

It is well known that, iterative learning control can handle any repetitive control process [1, 2]. Most of applications we can find in manipulators control in industry.

Some problems have iterative (or semiterative) character, but more than one output sequence is required. Consider problem of line following robot. To control robot speed, it is necessary to know curve shape. If we can compute associative memory, where input is line shape and output is required speed, we can handle this problem.

### 1. Problem formalisation

Consider two wheels (two motors) robot with differential drive [3]. For motors control signals we can write

$$r(n) = v(n) - d(n), \quad l(n) = v(n) + d(n), \quad (1)$$

where  $r(n)$  ( $l(n)$ ) is right (left) motor control output,  $v(n)$  is common control input,  $d(n)$  is difference control input.

The value of  $d(n)$  can be computed with sufficient precision using PD controller [5]. Our goal is to properly estimate  $d(n)$  which is speed of robot. Robot motion equations can be written as

$$\begin{aligned}\theta(n) &= (1 + \alpha)\theta(n - 1) - \alpha\theta(n - 2) + b_0(r(n) - l(n)) \\ \nu(n) &= \alpha\nu(n - 1) + (1 - \alpha)b_1(r(n) + l(n))\end{aligned}\quad (2)$$

where

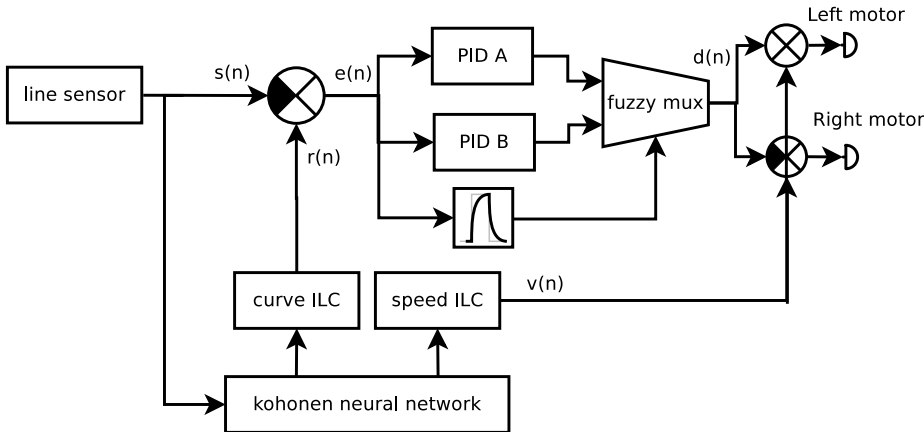
$\theta$  is robot orientation (Yaw angle),  $\nu$  is robot speed,  $\alpha$  is robot inertia constant, within  $(0, 1)$ ,  $b_0$  and  $b_1$  are constants dependent on wheel distances, wheel diameters, gear ratio, and maximum speed.

If required  $\theta$  and  $\nu$  are known, we can successfully use pure PID control defining error as  $e_\theta(n) = \theta_r(n) - \theta(n)$ ,  $e_\nu(n) = \nu_r(n) - \nu(n)$ , where  $\theta_r(n)$  and  $\nu_r(n)$  are required values. In our line following problem,  $\nu_r(n)$  is unknown, value  $\theta(n)$  is readed from line position sensor. In fact, inertia of motors is much smaller than inertia of whole robot, for control  $\theta(n)$  we can still use PD controller. Let the required value be  $\theta_r(n) = 0$ , which means robot is straight on line.

Our goal is to properly estimate  $v(n)$  and control robot speed, to give enough time for PD control to setup  $\theta(n)$ . We are minimizing  $e_\theta(n)$  and maximizing  $\nu(n)$  (coresponding to  $v(n)$ ).

## 2. Controller design

On figure 1 we can see block diagram of proposed controller.



**Figure 1.** Robot controller.

Input is line position from sensor  $s(n)$ . All values are normalised into  $\langle -1, 1 \rangle$ . Required value of  $\theta(n)$  is marked as  $r(n)$ . In simplified case, it can be set to 0. Computed error is used as input into two PD controllers (PID in general, I-term is equal to zero, because 2 ( $\theta$  part) has pole on  $1 + 0i$ . If I-term is nonzero, there will be second order pole on  $1 + 0i$  and system will be unstable).

PD controller outputs are mixed using fuzzy mux, which works as

$$d(n) = s'(n)a(n) + (1 - s'(n))b(n) \quad (3)$$

where  $a(n)$  is PID A output,  $b(n)$  is PID B output,  $s'(n)$  is select input, from interval  $\langle 0, 1 \rangle$ .

Select output is produced by nonlinear low pass filtered signal of  $e(n)$  as

$$s'(n) = \begin{cases} s'(n-1)k + (1-k)|e(n)|, & \text{for } |e(n)| < s'(n-1), \\ |e(n)|, & \text{otherwise,} \end{cases} \quad (4)$$

where  $k$  is filter constant, from  $(0, 1)$ . This filter is smoothing error values.

This system properly switches between two controllers, one for straight line and one for curved line.

To estimate  $v(n)$  we use Kohonen neural network [4] and iterative learning control. Neural network input is vector of  $s(n)$  values (implemented as FIFO, in experiments with size  $M=16$ ). Mark this vector as  $S(n) = [s(n), \dots, s(n-M)]$ .

Our goal, is to train network to make the classification of vectors  $S(n)$ . Depending on count of classification classes, we choose corresponding neurons count  $N$ . In experiments chooses we take  $N = 16$  which means, 16 different curve shapes can be recognized.

Transfer function of  $j$ -th neuron can be written as

$$y_j(n) = \sum_{i=0}^{M-1} |s(n-i) - w_j(i)| \quad (5)$$

where  $w_j$  are neuron weights, initially chosen randomly and modified during learning process.

Neuron with smaller  $y_q(n)$  will be marked as winning neuron, and weights modification will be done by formulae

$$w_q(n) = \eta w_q(n-1) + (1 - \eta)S(n), \quad (6)$$

where  $\eta$  is learning rate, from  $(0, 1)$  close to 1. Weights of neurons are slowly adapted to corresponding pattern. This can be illustrated on figure 2. Input was 2 dimensional, and goal is to find centers (green) of some data set (red). Input data set was produced by some Markov process. On figure 3  $y_q(n)$  values are shown. After found winning neuron, we need to find corresponding  $v(n)$ , denote it by  $v_q(n)$ .

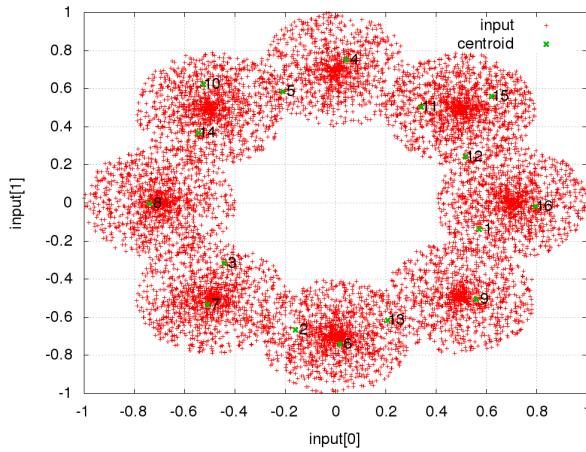


Figure 2. Kohonen test.

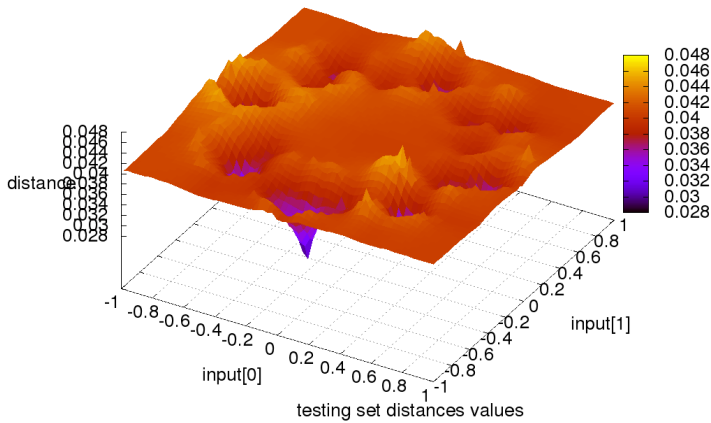


Figure 3. Kohonen test distances.

$$v_q(n) = v_q(n-1) + 1 - k s_f(n), \quad (7)$$

There  $s_f(n)$  is low pass filtered  $s(n)$  value. That means robot speed  $v(n)$  on curve type  $q$  is rising if  $s_f(n)$  is low.

### 3. Experimental results

Robot was learning for few loops on short line race with different curve types. Resulting curve types are on figure 4. Each line represents one curve shape. To reduce space in memory, absolute value of  $s(n)$  as input into Kohonen neural network is used. We can see many straight line positions, and few curves.

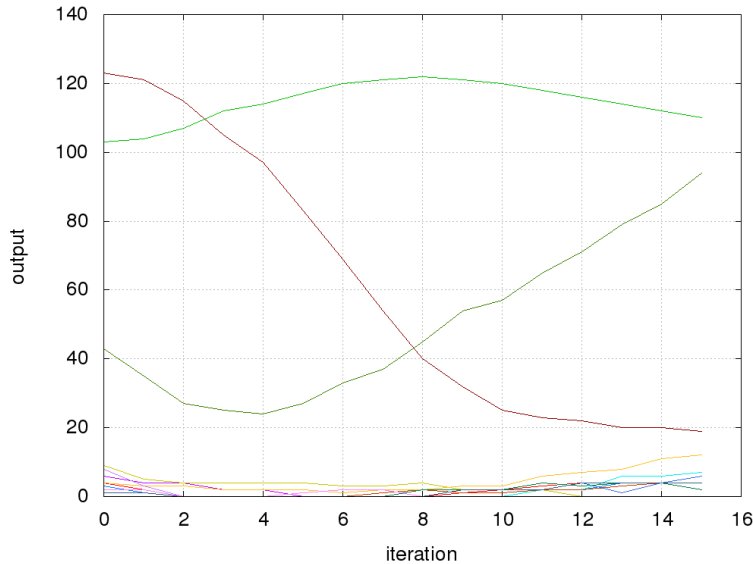


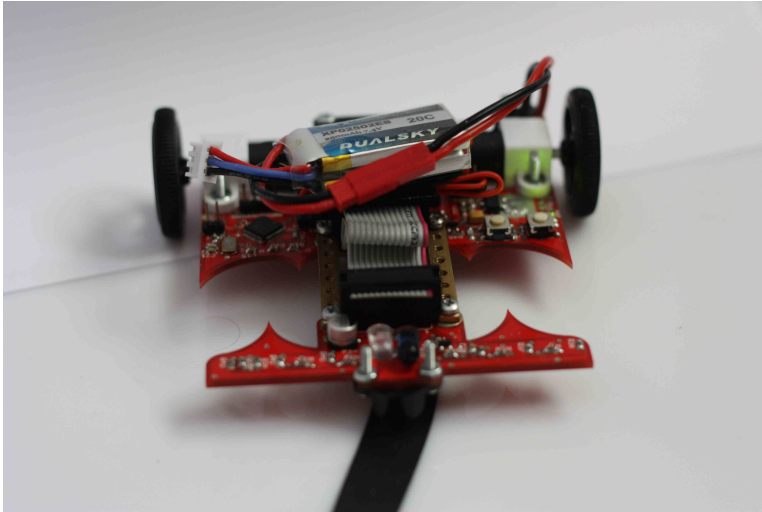
Figure 4. Resulting curves types.

### 4. Conclusions

In paper learning system for curve shape recognition using Kohonen neural network was described. Neural network results have corresponding speed output which control robot optional speed. Learning process is working in real time, on 75MHz ARM Cortex M4F microcontroller. Robot is on figure 5. Robot working video can be seen on [6]. More pictures are available on authors blog [7] and sources on github [8].

### References

- [1] Kevin L. Moore: Iterative learning control <http://inside.mines.edu/~kmoore/survey.pdf>.
- [2] Kevin L. Moore: An Introduction to Iterative Learning Control <http://inside.mines.edu/~kmoore/ilc-intro-pdftex03-03.pdf>.



**Figure 5.** Testing robot.

- [3] Patrícia N. Guerra: Linear modelling and identification of a mobile robot with differential drive <http://www.dca.ufrn.br/~adelardo/artigos/ICINC004b.pdf>.
- [4] R. Rojas: Neural Networks, Springer-Verlag, Berlin, 1996, Kohonen Neural Networks <http://page.mi.fu-berlin.de/rojas/neural/chapter/K15.pdf>
- [5] Michael Goldfarb, Taweedej Sirithanapipat: The effect of actuator saturation on the performance of PD-controlled servo systems <http://research.vuse.vanderbilt.edu/cim/pubs/journal/35%20-%20Goldfarb%20and%20Sirithanapipat.pdf>.
- [6] robot video link <https://www.youtube.com/watch?v=SGa1lbkyQMg>.
- [7] authors blog <http://aikenshin.blogspot.sk/>.
- [8] authors github [https://github.com/michalnand/motoko\\_after\\_math\\_linefollower](https://github.com/michalnand/motoko_after_math_linefollower).

## Contact address

**Ing. Michal Chovanec**, Department of Technical Cybernetics, Faculty of Management Science and Informatics, University of Žilina, Univerzitná 8215/1, 010 26 Žilina, Slovak Republic,  
*E-mail address:* [michal.chovanec@yandex.com](mailto:michal.chovanec@yandex.com), <http://aikenshin.blogspot.sk>



## MAPY V $\LaTeX$ -OVÝCH DOKUMENTOCH – PREDSTAVENIE BALÍČKA `getmap`

ALEŠ KOZUBÍK

**Abstrakt.** Cieľom príspevku je predstavenie balíčka `getmap`. Tento balíček umožňuje do  $\LaTeX$ -ových dokumentov zaradiť mapové materiály získané z externých zdrojov, ako sú OpenStreetMap alebo Google Maps a to aj s podporou Google Street View. V najjednoduchšom prípade pritom postačí aj špecifikácia požadovanej adresy. Balíček pre sťahovanie máp používa externý Lua skript, ktorý si vyžaduje aktiváciu funkcie `\write18`. Tento skript môže byť použitý aj samostatne z príkazového riadku.

**Kľúčové slová.** Maps,  $\LaTeX$ , `getmap`.

### MAPS IN THE $\LaTeX$ DOCUMENTS INTRODUCTION OF THE `getmap` PACKAGE

**Abstract.** The aim of this article is to introduce of the `getmap` package. This package allows to include into the  $\LaTeX$  documents the map materials obtained from the external resources such as OpenStreetMap and Google Maps and even with the support of Google Street View. In the simplest case, the specification of an address is sufficient. The package loads the map using the `\write18` feature, which must be activated to use this package. The image will be downloaded by an external Lua script that can be used also from the command line.

**Keywords.** Maps,  $\LaTeX$ , `getmap`.

## Úvod

Poznáme to všetci – občas treba do dokumentu zaradiť obrázky, ktorých obsahom je mapa určitej oblasti alebo lokality, prípadne aj s vyznačením trasy. Nieкто sa s týmto problémom stretáva častejšie (ak sa napríklad venuje geografii alebo pripravuje materiály pre cestovateľov), iný zriedkavejšie napríklad pri príprave pozvánok a pokynov pre konferencie. Ale určite sa s touto úlohou stretol každý. My si v tomto príspevku predstavíme užitočný balíček `getmap`, ktorý umožňuje priamo do  $\LaTeX$ ového dokumentu zakomponovať obrázky z takých zdrojov ako sú OpenStreetMap alebo Google Maps včítane vkladania obrázkov z Google Street View.

Podstata činnosti balíčka `getmap`, ktorého autorom je Josef Kleber, je pomerne jednoduchá. Balíček vlastne obsahuje jeden jediný príkaz `\getmap`, ktorý prostredníctvom jednoduchého Lua skriptu zabezpečí stiahnutie požadovaného

obrázku, teda mapy alebo fotografie z Google Street View do špecifikovaného grafického súboru. Takto získaný súbor potom vložíme do L<sup>A</sup>T<sub>E</sub>Xového dokumentu pomocou `\includegraphics`. Zložitejšie ako samotný príkaz sú teda jeho voliteľné argumenty, ktoré špecifikujú požadovaný obrázok. V článku sa teda budeme venovať týmto voliteľným položkám.

Pre správnu činnosť príkazu `\getmap` je potrebné aktívovať funkciu `\write18`, čo pri použití T<sub>E</sub>XLive znamená kompilovanie s prepínačom `--shell-escape` resp. pri použití MiKTeXu `--enable-write18`. Nakoľko mapy vkladáme do textu ako obrázky, je taktiež potrebné načítať balíček `graphicx`.

## 1. Príkaz `\getmap`

Ako sme už spomenuli, balíček `getmap` obsahuje defacto jediný príkaz, ktorého plná syntax má tvar:

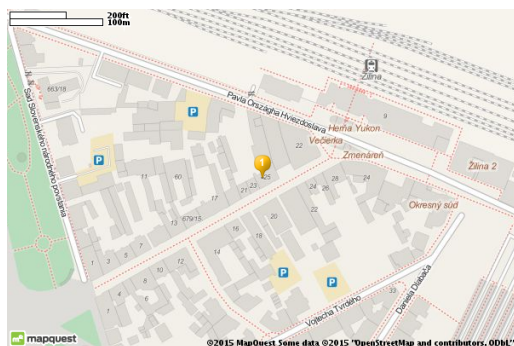
```
\getmap[volby]{adresa}
```

V najjednoduchšom prípade stačí poznať len adresu, POI alebo geografické GPS súradnice zvoleného miesta. Položka `adresa` musí byť plne rozvinutá a nesmie obsahovať žiadne makrá. Získaný obrázok sa implicitne ukladá do súboru s názvom `getmap.png`, ktorý sa je uložený v aktuálnom pracovnom adresári. V prípade, že do nášho dokumentu vkladáme len jednu mapu je tento mechanizmus plne postačujúci.

Príslušný zdrojový kód by teda mohol vyzeráť napríklad takto:

```
\getmap{Národná 25, 01001 Žilina, Slovakia}
\includegraphics[width=.5\linewidth]{getmap}
```

s výsledkom zobrazeným na obrázku 1.



**Obr. 1.** Ukážka výstupu získaného pomocou `\getmap`. Zobrazená je náhodne vybraná adresa v centre mesta Žilina

Pri opätovnej kompilácii súboru je potrebné si uvedomiť, že nedochádza ku prepisovaniu získaného obrázku `getmap.png`. To síce urýchľuje kompiláciu, na druhú stranu, ak dôjde ku zmene v zobrazovanej adrese to má za následok, že

obrázok v dokumente sa vlastne nezmení. To je možné riešiť buď odstránením súboru `getmap.png` z pracovného adresára pred novou kompiláciou, alebo vhodnou voľbou pre prepisovanie súborov, o ktorej budeme hovoriť neskôr.

## 2. Voľby príkazu `\getmap`

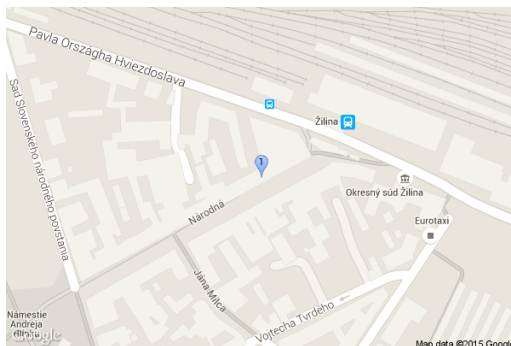
Činnosť príkazu `\getmap` podstatným spôsobom ovplyvňujú voliteľné argumenty. Dokonca je možné povedať, že sú dôležitejšie než samotný príkaz. My si predstavíme iba tie najdôležitejšie resp. najpoužívanejšie z nich, kompletný zoznam je možné nájsť v manuáli [3].

### Voľba `mode`

Voliteľný argument `mode` môže nadobúdať jednu z troch hodnôt `osm|gm|gsv` a určuje, ktorý mapový zdroj bude použitý. Z hodnoty argumentu je intuitívne zrejmé, ktorý zodpovedá OpenStreetMap, Google Maps alebo Google Street View. Ako implicitný zdroj je preddefinovaný OpenStreetMap. Ak by sme teda chceli výsledok na obrázku 1 získať pomocou Google Maps, je potrebné príkaz upraviť takto:

```
\getmap[mode=gm]{Národná 25, 01001 Žilina, Slovakia}
```

Výsledok si môžeme pozrieť na obrázku 2 .



**Obr. 2.** Ukážka výstupu na obrázku 1 získaného z Google Maps pomocou voľby `mode=gm`. Zobrazená je rovnaká adresa v centre mesta Žilina

### Voľby `file` a `imagetype`

Voľba `file=subor` umožňuje pomenovať výstupný súbor, ktorý bude vytvorený príkazom `\getmap`. Názov súboru sa uvádza bez prípony. Pre stanovenie grafického formátu získaného výstupu sa potom používa voľba `imagetype`, ktorá môže nadobúdať niektorú z hodnôt `png|jpeg|jpg|gif`, pričom ako implicitný typ je

súbor vo formáte `png`. Táto voľba už je ale viazaná na použitý mód, pričom uvedené hodnoty sú dostupné pre `mode=osm`. Pri použití módu `gm` sú dostupné ešte ďalšie grafické formáty ako napríklad `png8|png32|`.

Ak si teda uvedomíme, že pri spracovaní obrázku 1 bol použitý implicitný názov súboru `getmap`, je zrejmé, že súbor vložený do obrázku 2 je potrebné aj premenovať. Úplný zdrojový kód pre výstup na obrázku 2 teda vyzerá takto:

```
\getmap[mode=gm,file=myobr]{Národná 25, 01001 Žilina, Slovakia}
```

### Voľba `type`

Voľba `type` môže v móde `osm` nadobúdať niektorú z troch hodnôt `map|sat|hyb` pre zobrazenie mapy, satelitnej snímky alebo hybridné zobrazenie mapy a satelitnej snímky. Pri nastavení módu na hodnotu `gm` sa možné hodnoty voliteľného argumentu `type` menia na `roadmap|satellite|hybrid|terrain`.

Na obrázku 3 ilustrujeme satelitný záber rovnakého výrezu centra mesta ako na mapách na obrázkoch 1 a 2.



**Obr. 3.** Ukážka výstupu rovnakej oblasti ako na obrázku 1, získaného z Google Maps pomocou voľby `type=satellite`. Zobrazená je rovnaká adresa v centre mesta Žilina

### Voľba `overwrite`

Ide o dôležitý voliteľný argument, ktorý určuje, či sa pri jednotlivých kompiláciách budú získané obrázky prepisovať alebo nie. Nastavuje sa na logické hodnoty `true|false`. Tento argument je užitočný, ak meníme definovaný výrez z mapy, aby sme nemuseli manuálne odstraňovať obrázky z predchádzajúcich kompilácií. Naopak, jeho nastavenie na hodnotu `false` urýchľuje preklad, ak máme obrázkov veľa a sú už z predchádzajúcich behov  $\LaTeX$ u vytvorené.

### Voľba `xsize` a `ysize`

Tieto voliteľné hodnoty určujú rozmery získaného obrázku v pixeloch, pričom hodnota `xsize` určuje jeho šírku a hodnota `ysize` určuje jeho výšku. Implicitné

rozmery sú nastavené na hodnoty  $600 \times 400$ , pričom hodnotu je možné meniť. V režime `osm` pre oba rozmery horné hranice 3840, kým slobodná verzia Google maps je obmedzená na rozmery  $640 \times 640$ .

### 3. Vyznačovanie objektov na mapách

Pre vyznačenie významných objektov či orientačných bodov na mape slúžia tzv. markery. Tieto sa definujú pomocou voliteľného argumentu `markers` a ako hodnota je mu priradený zoznam všetkých značiek, ktoré majú byť vyznačené na mape. Zoznam markerov sa uzatvára do zložených zátvoriek, pričom každý z nich je definovaný sekvenciou:

```
&markers=size:mid|color:blue|label:S|loc1|loc2|
```

Jednotlivé položky potom definujú vlastnosti markerov. Hodnota parametra `size` môže nadobúdať hodnoty `tiny`, `mid`, `small`, pričom ako implicitná hodnota sa berie `mid`. Parameter `color` definuje farbu markera, ktorá môže byť zadaná buď pomenovaním alebo hexadecimálnym kódom. parameter `label` určuje značku vo vnútri markera (len pri veľkosti `mid`). Prípustné sú numerické znaky alebo kapitálky. Hodnoty `loc1|loc2` potom definujú lokalizáciu markera na mape.

Ukážku mapy s vyznačenými objektmi vidíme na obrázku 4, kde je zobrazené okolie Fakulty Riadenia a Informatiky a v jej blízkosti dve reštaurácie. Príslušný kód príkazu `\getmap` vyzerá takto:

```
\getmap[file=fricka,overwrite=true,mode=gm,
markers={&markers=size:mid|label:F|color:red|
  Fakulta riadenia a informatiky, 01008 Žilina, Slovakia%
&markers=size:mid|label:M|color:blue|Obchodná 3269,01008 Žilina, Slovakia,%
&markers=size:mid|label:K|color:blue|Slovanská 3278,01008 Žilina, Slovakia},%
visible={{49.16, 18.65}}|{49.21, 18.45}}
]{}
```

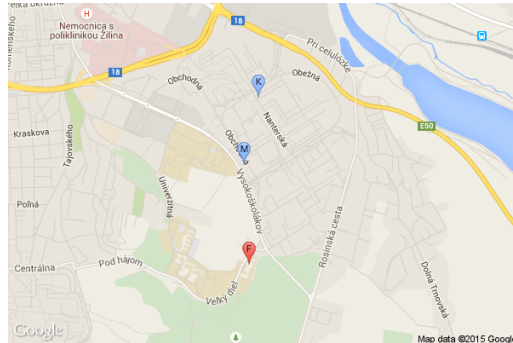
V zdrojovom kóde si môžeme všimnúť tiež argument `visible`. Jeho úlohou je zabezpečenie zobrazenia takého výrezu mapy, aby všetky lokality uvedené v zozname a oddelené zvislou čiarou boli na obrázku viditeľné. Je ich možné zadať ako objekty alebo pomocou GPS súradníc.

### 4. Vyznačenie trasy

Často je potrebné v mape vyznačiť aj cestu, ako je možné prejsť od jedného objektu ku druhému, prípadne viacero trás. Pre tieto účely má príkaz `\getmap` voliteľný parameter `path`, ktorý podobne ako `markers` očakáva zoznam URL parametrov oddelených zvislou čiarou v tvare:

```
&path=weight:5|color:orange|loc1|loc2|
```

Hodnota `weight` udáva hrúbku čiary vyznačujúcej trasu, `color` farbu tejto čiary a ostatné hodnoty predstavujú lokalizačné údaje. Lokalizačné údaje je pri tom

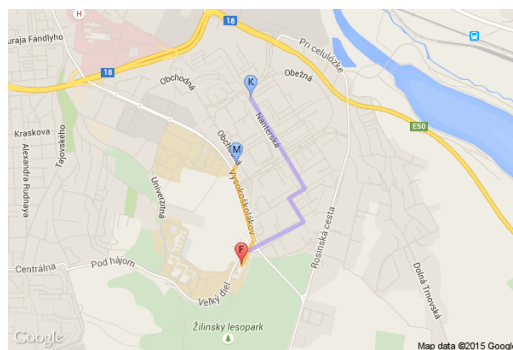


**Obr. 4.** Ukážka mapy okolia FRI ŽU (marker F) s vyznačením stravovacích zariadení „Miláno“ (marker M) a „Kazačok“ (marker K)

možné zadávať buď ako adresy alebo dvojicu GPS súradníc. Vyznačenie dvoch trás je ilustrované na obrázku 5.

Príslušný zdrojový kód upravíme takto

```
\getmap[file=fricka3,overwrite=true,mode=gm,
markers={&markers=size:mid|label:F|color:red|Fakulta riadenia a informatiky,
01008 Žilina, Slovakia%
&markers=size:mid|label:M|color:blue|Obchodná 3269,01008 Žilina, Slovakia,%
&markers=size:mid|label:K|color:blue|Slovanská 3278,01008 Žilina, Slovakia},%
path={&path=weight:5|color:orange|Fakulta riadenia a informatiky, 01008 Žilina,
Slovakia|49.203912,18.763293|49.208426,18.761190|
Obchodná 3269,01008 Žilina, Slovakia%%
&path=weight:5|color:purple|Slovanská 3278,01008 Žilina, Slovakia|49.207304,
18.768442|49.206660,18.766833|49.205117,18.768314|49.203267,18.762885}] {}
```



**Obr. 5.** Ukážka mapy okolia FRI ŽU (marker F) s vyznačením stravovacích zariadení „Miláno“ (marker M) a „Kazačok“ (marker K) a trasy ku nim

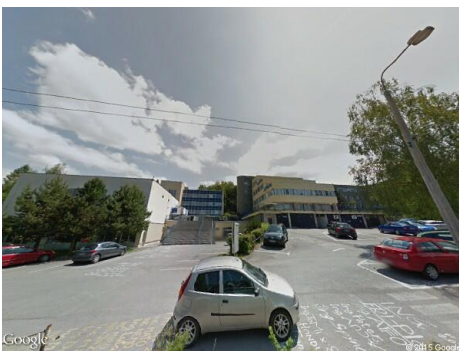
## 5. Obrázky Google Street View

Balíček `getmap` umožňuje vkladať do dokumentu aj fotografie, získané prostredníctvom služby Google Street View. Pre tieto účely je potrebné pomocou voliteľných argumentov nastaviť službu `mode=gsv`. Pre špecifikáciu záberu sú dôležité voľby `heading`, ktorá udáva smer záberu zo stanoveného miesta v rozpätí 0–360 (pre sever použijeme hodnotu 0, pre východ 90 atď.), `pitch`, ktorý definuje uhol sklonu kamery od vodorovnej roviny a udáva sa v rozpätí od –90 po 90 a voľba `fov` udávajúca veľkosť rozsahu horizontálneho záberu v stupňoch a to v rozsahu 0–120. Všetky uvedené parametre sú merané v stupňoch.

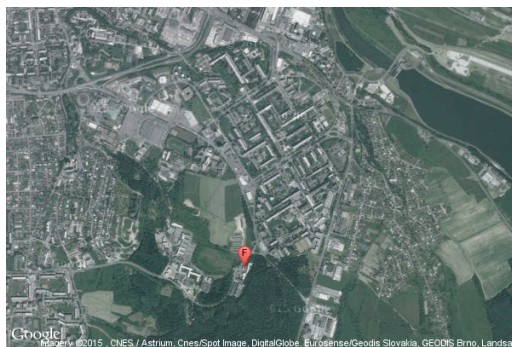
Ilustrujeme si to na obrázku fakulty. Príslušnú snímku získame napríklad takýmto príkazom:

```
\getmap[file=pfricka,mode=gsv,heading=120,fov=120,pitch=20,xsize=600,ysize=450,scale=2,zoom=20,overwrite=true]{49.202217,18.761407}
```

Výsledok potom vidíme na obrázku 6. Na susednom obrázku 7 vidíme satelitný záber okolia fakulty s vyznačeným bodom snímania.



**Obr. 6.** Snímka FRI zo služby Google Street View



**Obr. 7.** Satelitný záber okolia FRI ŽU

## 6. Script `getmapdl`

Ako sme sa už zmienili v úvode tohto príspevku, balíček `getmap` vlastne využíva jeden Lua skript, ktorý môže byť použitý aj z príkazového riadku. Kompletnú náponedu ku skriptu získame z príkazového riadku obvyklým postupom, teda:

```
$ getmapdl -h
```

Z dostupných prepínačov si uvedieme len niekoľko najdôležitejších:

- m určuje mód, obvykle `osm|gm|gsv`,
- l určuje lokalitu,
- i určuje mód formátu grafického súboru,
- o určuje pomenovanie výstupného súboru (bez prípony).

Ak to zhrnieme, potom pomocou príkazu

```
getmapdl -m gm -l 'Národná 25,01001 Žilina, Slovakia' -i jpg -o zilina
```

získame rovnaký obrázok, ako je na obrázku 2 a výsledok bude uložený v súbore `zilina.jpg`.

## Záver

V príspevku sme si predstavili balíček `getmap`. Na ukážkach bolo demonštrované, ako je do textu možné zaradiť nielen mapy získané zo služieb Open Street Map alebo Google Maps, ale aj fotografie objektov získané pomocou Google Street View.

**Podakovanie.** Tento príspevok vznikol s láskavým príspevom grantu KEGA–011ŽU–4/2014 „Experimentálna matematika – zviditeľnenie neviditeľného“ podporeného Slovenskou kultúrno-edukačnou grantovou agentúrou.

## Literatúra

- [1] BLAŠKO, R.: *L<sup>A</sup>T<sub>E</sub>X nie je farba na maľovanie*, Otvorený softvér vo vzdelávaní, výskume a IT riešeníach, zborník medzinárodnej konferencie OSSConf 2010, Žilina, 1.–4. júla 2010, str. 43–52, ISBN 978-80-970457-0-8, <http://ossconf.soit.sk/images/zborniky/zbornik2010.pdf>.
- [2] BLAŠKO, R.: *L<sup>A</sup>T<sub>E</sub>X nie je farba na maľovanie, ale na písanie*, Otvorený softvér vo vzdelávaní, výskume a IT riešeníach, zborník medzinárodnej konferencie OSSConf 2011, Žilina, 1.–4. júla 2011, str. 249–258, ISBN 978-80-970457-1-5, <http://ossconf.soit.sk/images/zborniky/zbornik2012.pdf>.
- [3] KLEBER, J.: *Downloading maps from OpenStreetMap, Google Maps or Google Street View*.
- [4] KOPKA, H. – DALY, P. W.: *L<sup>A</sup>T<sub>E</sub>X – Podrobný príručce*, Brno, Computer Press, 2004, ISBN 80-722-6973-9.
- [5] KOZUBÍK, A.: *Prezentačné materiály v triede Bemer*, Zborník príspevkov z medzinárodnej konferencie OSSConf 2011, Žilina, 1.–4. júla 2011, str. 223–235, ISBN 978-80-970457-1-5.
- [6] RYBIČKA, J.: *L<sup>A</sup>T<sub>E</sub>X pro začátečníky*, Brno, KONVOJ 2003, ISBN 80-7302-049-1.
- [7] RYBIČKA, J., ČAČKOVÁ, P., PŘICHYSTAL J.: *Průvodce tvorbou dokumentů*, Bučovice, Nakladatelství Martin Stříž 2011, ISBN 978-80-87106-43-3.
- [8] STRÍŽ, P.: *Sazba v T<sub>E</sub>Xu a kresba v METAPOSTu*, Bučovice, Nakladatelství Martin Stříž 2011, ISBN 978-80-87106-51-8.

## Kontaktná adresa

**RNDr. Aleš Kozubík, PhD.**, Katedra matematických metód a operačnej analýzy, Fakulta riadenia a informatiky, Žilinská univerzita, Univerzitná 8215/1, 010 26 Žilina, Slovenská republika,

*E-mailová adresa:* [alesko@frcatel.fri.uniza.sk](mailto:alesko@frcatel.fri.uniza.sk)



## INSIGHT MAKER: SIMULÁCIA ÚDRŽBY A OBNOVY STROJOV

ALENA PEŠKOVÁ (SK)

**Abstrakt.** V tomto príspevku predstavujeme simulačný nástroj Insight Maker aplikovaný na problematiku údržby a obnovy strojov. Je vhodný aj pre používateľov, ktorí nemajú programátorské skúsenosti. Tento otvorený software umožňuje komunikáciu s tabuľkovými procesormi, kde sa dá s výsledkami zo simulácie ďalej pracovať. Možnosti Insight Makeru demonštrujeme na ilustračnom príklade dynamického odstraňovania porúch strojov. Navrhujeme plánovanie údržby v situáciách, kde odstránenie porúch má prednosť pred plánovanou preventívnou údržbou. Výsledkom simulácie sú grafy, ktoré popisujú vznik porúch a opráv strojov pri preventívnej údržbe.

**Kľúčové slová.** Insight Maker, dynamická simulácia, údržba, obnova.

## INSIGHT MAKER: SIMULATION OF MAINTENANCE AND RENEWAL OF MACHINES

**Abstract.** We present simulation tool Insight Maker in this paper applied to problems of maintenance and machines renewal. It is also suitable for users who do not have programming experience. This open software allows communication with the spreadsheets, where simulation results can be processed. We demonstrate the possibilities of Insight Maker in the illustrative example of dynamic elimination of machine failures. We propose to plan a maintenance in the situations when elimination of the failures has precedence over planned preventive maintenance. The results of simulations are graphs that describe the creation of failures and repairs of machines during their preventive maintenance.

**Keywords.** Insight Maker, dynamic simulation, maintenance, renewal.

## Úvod

Pre používateľov simulačných nástrojov, ktorí nemajú skúsenosti s programovaním je potrebné hľadať čo najjednoduchší program, ktorý tento nedostatok odstraňuje. Do úvahy prichádza grafický program Vensim [5]. V zahraničí vznikli viaceré články a práce, kde autori našli uplatnenie tohto programu pri riešení problematiky údržby. Spomeňme dizertačnú prácu [1] „Modelovanie systémov priemyselnej údržby a efekty podmieneného monitorovania“, kde sa rieši údržba zo systémového hľadiska. Hlavné výsledky potvrdzujú, že subjektívny charakter zlyhania, dostupné informácie o strojoch ako aj opakovanosť údržby sú najdôležitejšie faktory v riadení systému údržby. Autor efektívne využíva Vensim pri

modelovaní rozdelenia pracovníkov pri údržbe, odstraňovaní porúch komponentov, kontroly preventívnej údržby i modelovaní dodávateľských reťazcov. Program je ale komerčný, voľne dostupný je len na obmedzený čas.

V práci [2] sme sa venovali možnosti simulácie obnovy a údržby v Petriho sieťach. Pokúšali sme sa experimentovať s otvorenými softvérmi Snooply a Pipe2. Ukázalo sa, že ich zvládnutie vyžaduje značnú znalosť o simulačných technológiach. I to nás viedlo k voľbe simulačného nástroja Insight Maker [4].

## 1. Simulačný nástroj Insight Maker

Insight Maker je grafický nástroj pre modelovanie a simuláciu, ktorý bol uverejnený na konci roku 2009. Odvtedy prešiel neustálym vývojom a získal už viac ako 20 000 registrovaných používateľov. Spätná väzba od týchto používateľov pomáha vylepšovať a vyvíjať tento nástroj. Jeho prednosťou je, že ho netreba inštalovať, ale možno s ním pracovať vo webovom prehliadači (Google Chrome, Mozilla, atď).



Obr. 1. Uživatelské prostredie v Insight Maker

Podmienkou pre prácu s programom je vytvorenie používateľského konta. Navrhnuté simulácie sa dajú uložiť alebo publikovať. Publikovanú simuláciu je možné konzultovať s ďalšími používateľmi tohto programu. Doterajšia oblasť využitia simulačného programu je široká. Nájsť sa tam dajú uverejnené simulácie z ekonomiky, bankovníctva, zdravotníctva, teórie zásob, riešenia výrobných problémov, atď. Možno vytvoriť skupiny podľa tém záujmu do ktorým sa noví používatelia môžu zapojiť. Simulačný nástroj ponúka aj tutoriály pre rýchle oboznámenie sa s programom. Veľkou výhodou je, že ak nemáme veľa skúsenosti s týmto programom, v prípade potreby sa môže obrátiť na skúsenejších používateľov (forum, mail). Insight Maker je vhodný nástroj hlavne pre tých, ktorí pracujú

s Microsoft Excel, LibreOffice Calc alebo iným tabuľkovým procesorom. Výstupy zo simulácie sa dajú exportovať v tvare \*.csv a následne sa dá s takýmto súborom ďalej pracovať.

## 2. Simulačný model obnovy a údržby strojov

Simulačný nástroj predstavíme na ilustračnom príklade. Obmedzíme sa len na niektoré základné prvky v Insight Maker, ktoré sú nevyhnutné pri simulácii dynamických systémov.

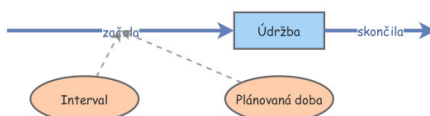
### 2.1. Formulácia problému

Majme situáciu, kde v danom sledovanom období treba dynamicky simulovať činnosti pri obnove a údržbe strojov v nepretržitej prevádzke:

*Je daný počet opravárov a počet strojov, na ktorých treba vykonávať pravidelnú údržbu strojov a odstraňovať poruchy. Na údržbu strojov je vyčlenený denne jeden pracovník a vykonáva ju v pravidelných plánovaných intervaloch vopred známu dobu. Keď pracovník nemá naplánovanú údržbu, je nečinný, pokiaľ sa nevyskytne porucha. Pracovníci sú navzájom zastupiteľní. Neuvažuje sa s prestávkami na od-dych pracovníkov. Poruchy sa odstraňujú, čo najskôr po vzniku, pričom je známa intenzita ich vzniku a priemerná doba na jej odstránenie (náhodné udalosti). Uva-žuje sa taká stratégia kooperácie činnosti údržby a obnovy, kde odstránenie poru-chy má prednosť pred pravidelnou preventívnou údržbou, ktorá sa môže vynechať. Výsledkom simulácie by mal byť časový vývoj ukazovateľov:*

- počet voľných opravárov,
- počet realizovaných opráv strojov,
- počet preventívnych údržieb strojov.

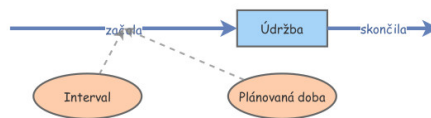
V simulácii sme zvolili sledované obdobie 10 dní t. j. 240 hodín, počet opravá-rov 5, počet strojov 10, plánovaný interval preventívnej údržby 4 dni (96 hod.) v trvaní 3 hodín, vznik porúch je generovaný Poissonovským tokom s priemernou medzerou medzi poruchami 5 dní (120 hod.), priemerná doba na odstránenie po-ruchy je 3 hod. a je modelovaná exponenciálnym rozdelením. Vidíme, že čas je vhodné merať v hodinách.



Obr. 2. Simulačný diagram v modeli údržby

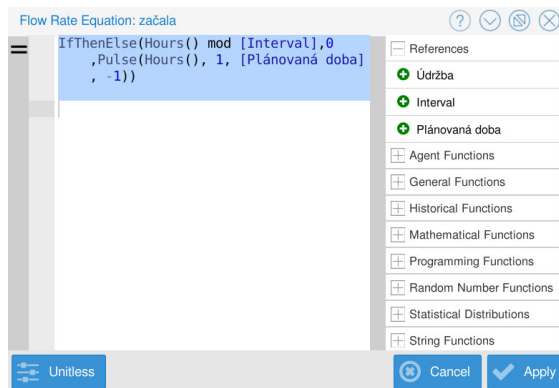
## 2.2. Model údržby

Najskôr budeme modelovať jednoduchý deterministický model údržby s jedným údržbárom. Potrebujeme k tomu dva prvky, v **Insight Maker** nazývané **Primitive** a to, **Stock** (Sklad) a **Variable** (Premenná). Na obr. 3 máme diagram, ktorý obsahuje **Údržba** ako sklad, kde sa kumulujú poruchy a **Interval** udávajúci dobu medzi dvoma bezprostredne nasledujúcimi požiadavkami na údržbu. V premennej **Interval** máme uloženú konštantu veľkosti 96 [hod.] a v premennej **Plánovaná doba** máme uloženú konštantu 3 [hod.].



Obr. 3. Simulačný diagram v modeli údržby

Výpočet doby údržby na obr. 4 je už obtiažnejší. Potrebujeme totiž dosiahnuť, aby po každom násobku intervalu údržby bola vykonaná údržba stroja. Po simulácii dostaneme vývoj údržby jedného stroja na obr. 5, údržba sa opakuje pravidelne po 96 hodinách v časoch 0, 96 a 192 v trvaní troch hodín.

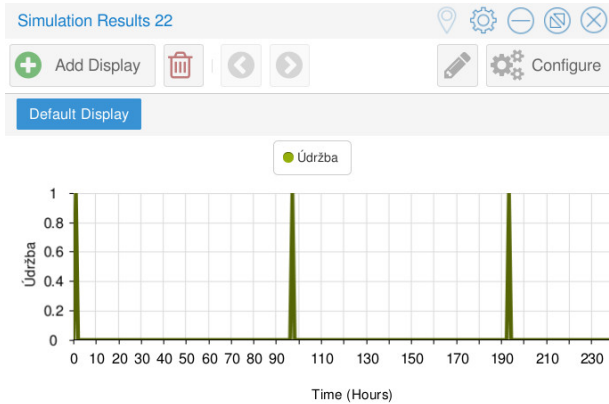


Obr. 4. Výpočet doby údržby

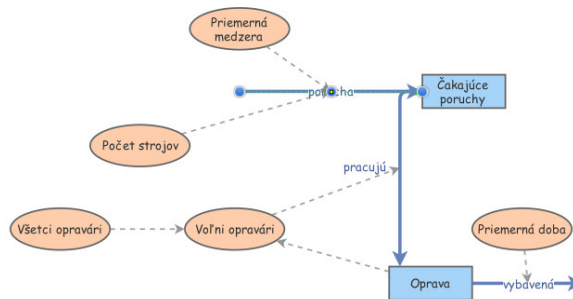
Ako vidíme v uvedenom deterministickom modeli, je obtiažnosť jeho tvorby sústredená na návrh grafu diagramu primitív s tokovými a informačnými prepojeniami (šípkami – orientovanými hranami) medzi nimi.

## 2.3. Model obnovy

V prípade obnovy strojov máme stochastický tok porúch strojov, dobu na ich odtrávenie ponecháme na začiatok konštantnú.



Obr. 5. Vývoj preventívnej doby údržby jedného stroja



Obr. 6. Simulačný diagram v modeli obnovy

Vstupný tok porúch na obr. 6 by bol pre jeden stroj určený Poissonovým rozdelením s intenzitou rovnou prevrátenej hodnote priemernej dĺžky medzery medzi poruchami, ktorú máme v premenennej **Priemerná medzera**. My však máme 10 strojov a tak musíme túto intenzitu násobiť ich počtom. Teraz už môžeme tieto dva nezávislé modely spojiť do jedného.

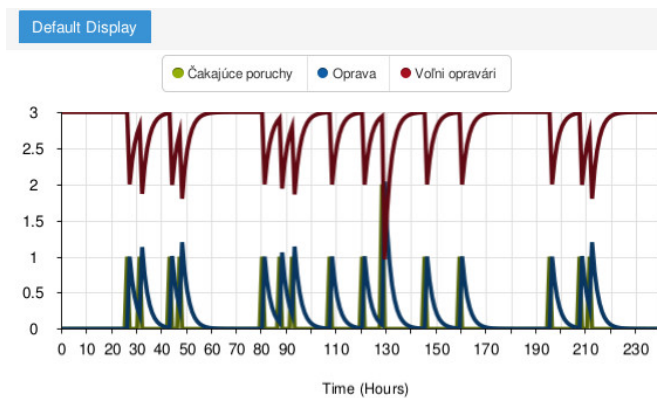
## 2.4. Model obnovy a údržby

V tomto modeli spojíme obnovu s údržbou a budeme navyše predpokladať, že aj doba opravy strojov je stochastická a je modelovaná exponenciálnym rozdelením.

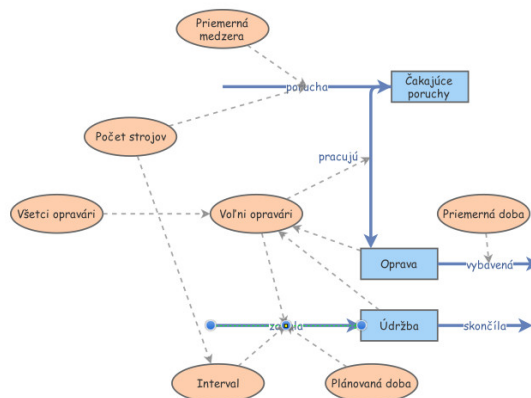
Údržba je tu ovplyvnená počtom strojov, na ktorých treba vykonávať preventívnu údržbu a počtom voľných opravárov. Asi najobťažnejším miestom v diagrame na obr. 9 je podmienka v toku, kde **začala údržba**. Vykonanie údržby je podmienené existenciou aspoň jedného voľného opravára v plánovanom čase preventívnej údržby. Príkaz na obr. 10 pomocou dvoch vnorených podmienok

| Flow                |  |
|---------------------|--|
| General             |  |
| (name)              | porucha  |
| Note                |  |
| Configuration       |  |
| Flow Rate =         | RandPoisson([Počet strojov]/[Priemerná medzera]) |
| Only Positive Rates | Yes  |
| Slider              |  |
| Show Value Slider   | No   |

Obr. 7. Generovanie Poissonovho toku porúch strojov v modeli obnovy

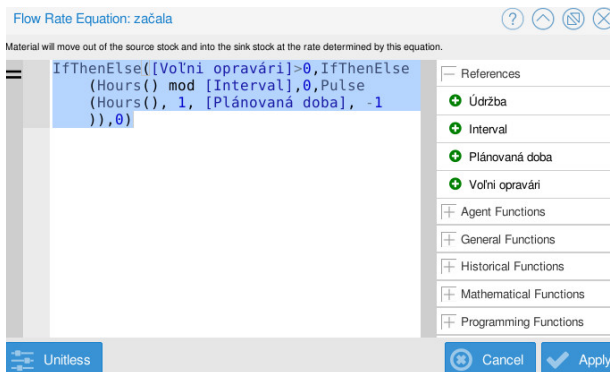


Obr. 8. Obrázok 7. Vývoj sledovaných ukazovateľov v modeli obnovy

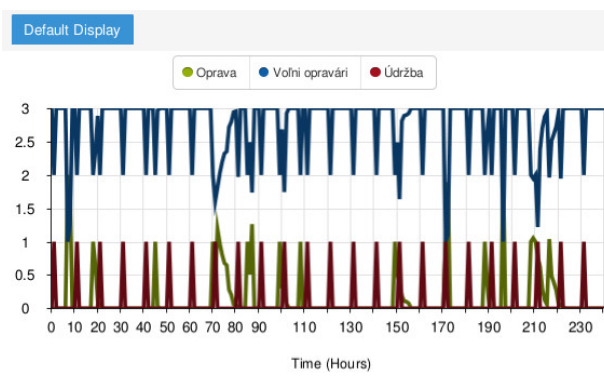


Obr. 9. Simulačný diagram modelu obnovy a údržby

IfThenElse() už vyžaduje istú programátorskú zručnosť. Modelovanie exponenciálnej doby obsluhy je analogické modelovaniu vstupného toku porúch. Po simulácii dostaneme, na obr. 11, graf vývoja požadovaných ukazovateľov činností.



Obr. 10. Príkaz pre začatie preventívnej údržby stroja voľným opravárom



Obr. 11. Vývoj sledovaných ukazovateľov v modeli obnovy a údržby

Uvedený graf vývoja ukazovateľov môžeme importovať do súboru v tvare tabuľky a následne štatisticky spracovať.

### 3. Záver

V tomto príspevku sme sa zaoberali len dynamickým modelovaním údržby a obnovy strojov pri prioritnej obnove strojov voči údržbe. V ďalších prácach sa plánujeme zamerať na analýzu rôznych stratégií koordinácie týchto činností z hľadiska ekonomickej efektívnosti. V prípade zapracovania spoľahlivostných ukazovateľov strojov tu bude potrebné zoznámiť sa aj s nástrojmi agentovej simulácie, ktorá už kladie väčšie požiadavky na používateľov.

Prvé skúsenosti so simulačným nástrojom *Insight Maker* nás utvrdili v presvedčení, že je to veľmi dobrá voľba pre používateľa s minimálnymi znalosťami

programovania. Oceňujeme množstvo dostupných príkladov na internete, z ktorých možno čerpať inšpiráciu, ako aj z priateľsky ladeného fóra používateľov **Insight Maker**-u. Skutočnosť, že je sa jedná o webovú aplikáciu, ktorá je nezávislá na webovom prehliadači, poteší nielen začiatočníkov.

## Literatúra

- [1] HONKANEN T.: *Modelling Industrial Maintenance Systems and the Effects of Automatic Condition Monitoring*, dissertation, Helsinki University of Technology, Information and Computer Systems in Automation, Espoo, 2004, 122p. <http://lib.tkk.fi/Diss/2004/isbn9512268167/isbn9512268167.pdf>
- [2] PEŠKOVÁ A.: *Modelling Maintenance and Renewal in Petri Nets*, In: Applied Mechanics and Materials : ROBTEP 2012, 11th International Conference on Industrial, Service and Humanoid Robotics, Štrbské Pleso, Slovakia, 14-16 November 2012, Vol. 282 (2013), pp. 282-286, ISBN 978-303785603-1, ISSN 1660-9336.
- [3] ROE S. F.: *Insight Maker: A General-purpose Tool for Web-based Modeling & Simulation, Simulation Modelling Practice and Theory* 47, 2014, pp. 28–45.
- [4] INSIGHT MAKER, <https://insightmaker.com>.
- [5] VENSIM, Ventana Systems, <http://Vensim.com>.

## Kontaktná adresa

**Ing. Alena Pešková**, Katedra výrobnéj techniky, Strojnícka fakulta, Technická univerzita v Košiciach, Némcovej 32, 042 00 Košice,

*E-mailová adresa:* [alena.peskova@tuke.sk](mailto:alena.peskova@tuke.sk)



## LMS CHAMILO V NIŽŠOM SEKUNDÁRNOM VZDELÁVANÍ

VLADIMÍR PILIŠ (SK) A LUBOŠ BALÁŽOVIČ (SK)

**Abstrakt.** Článok sa venuje problematike praktického využitia e-learningového softvéru Chamilo vo vyučovaní geografie v piatom ročníku základnej školy. Prvá časť v krátkosti opisuje problematiku digitalizácie vzdelávania a implementáciu open source nástrojov do vyučovacieho procesu na slovenských základných a stredných školách. Druhá časť stručne popisuje metodiku prospešnú pre učiteľov základných a stredných škôl – uvádza prehľad komplexných pracovných postupov pri konštrukcii interaktívnych pracovných listov, testov a učebnej osnovy. Záverečná tretia časť sa venuje výsledkom pilotného odskúšania elektronických učebných materiálov v praxi na základnej škole v priebehu školského roka 2014/2015. Výsledky testovania potvrdili ich praktickú uplatniteľnosť a tiež pomohli identifikovať rad technických a didaktických problémov ktorým je potrebné venovať ďalšiu pozornosť.

**Kľúčové slová.** Chamilo, e-learning, digitalizácia vzdelávania, vzdelávanie v geografii na ZŠ.

## LMS CHAMILO IN LOWER SECONDARY EDUCATION

**Abstract.** This paper is focused on the application of the practical use of Learning Management System (LMS) Chamilo in Geography teaching in the Year 5 of lower secondary education. The first part briefly summarises the digitalisation of education and implementation of open source tools in the learning process at primary and secondary schools in Slovakia. The second section describes a methodology that is beneficial to teachers of lower and higher secondary education. It provides an overview of complex workflows in the design of interactive worksheets, tests and curriculum. The final third part is devoted to the results of pilot testing of electronic teaching materials in practical lower secondary education during the school year 2014/2015. The test results confirmed the practical applicability and also helped to identify a number of technical and didactical problems which would need further attention.

**Keywords.** Chamilo, e-learning, education digitalization, lower secondary education of geography.

## Úvod

Elektronické učebné pomôcky vo veľmi značnej miere uľahčujú žiakom pochopenie konkrétnych učív, učiteľom zdĺhavé prípravy, zefektívňujú vyučovací proces, motivujú a aktivizujú žiakov, čím pomáhajú skvalitniť a prehĺbiť ich fantáziu. Overenie nových učebných pomôcok v praxi je nevyhnutné pri akejkoľvek novej učebnej pomôcke, avšak jeho realizácia nie je jednoduchá vzhľadom na rozmanitosť škôl, ich technické a personálne vybavenie. Výraznou prekážkou

plošného nasadenia elektronickej učebnej pomôcky je skutočnosť, že viaceré existujúce proprietárne e-learningové systémy neboli zatiaľ overené v našej školskej praxi. Problematiku učebných materiálov z geografie sme riešili v prostredí LMS Chamilo, ktorý v určitých podmienkach predstavuje lepšie využiteľnú alternatívu e-learningového systému vo vyučovaní geografie na základných školách. V tomto článku prinášame výsledky pilotného testovania programu Chamilo v podmienkach reálneho vyučovacieho procesu na základnej škole.

## 1. Digitalizácia vzdelávania na Slovensku

Súčasný vzdelávací proces je čoraz náročnejší na používanie moderných technológií, ktorých zvládnutie je predpokladom zvyšovania kvality vyučovacieho procesu. Počas uplynulých štrnástich rokov rezort školstva Slovenskej republiky spustil a realizoval množstvo projektov zameraných na modernizáciu školstva. Elektronické učebné materiály sa dnes stávajú rovnocenným partnerom klasických učebníc a pracovných listov. V niektorých štátoch (napr. Česká republika) sú vydávané ako príloha papierovej učebnice, či pracovného listu. Na Slovensku neexistuje ani jedno komplexné riešenie, ktoré by nadväzovalo na učebnice či poreformný štátny vzdelávací program. Najviac sa mu priblížil projekt *Planéta vedomostí*, ktorý bol zakúpený Ministerstvom školstva, vedy výskumu a športu SR koncom roka 2012 [2]. Jej nedostatkom okrem medializovaných chýb je, že nepokrýva napríklad ani čiastkovo učebný predmet geografia.

Čiastkové riešenia vzišli z komunitných portálov (napr. [11]), z komerčnej sféry (Edupage a jeho prémiový obsah) alebo z veľkých štátnych projektov (v minulosti projekt Infovek [9], projekt *Modernizácia vzdelávania na základných a stredných školách* [10]). V súčasnosti na príprave elektronických vzdelávacích materiálov pracuje projekt *Digitalizácia vzdelávania* [7]. Ani jeden z vyššie uvedených projektov nerátal s plošným nasadením opensource riešení napriek tomu, že existujú pozitívne príklady zo zahraničia.

Spoločným cieľom všetkých týchto projektov je snaha o pokrytie väčšiny predmetov vyučovaných na ZŠ a SŠ. Celková kapacita zdrojov je rozdelená tak, že obsahovo sa z geografie riešia len niektoré vybrané témy, pričom tieto fragmenty nemajú častokrát žiadnu spoločnú koncepciu. Druhým znakom týchto projektov je chýbajúce, alebo len veľmi povrchné overenie v praxi. Overenie v praxi, pokiaľ vôbec bolo realizované, bolo riešené jedným učiteľom v jednej škole a väčšinou iba v jednej jedinej triede. Ďalším faktorom charakterizujúcim aktuálny stav je relatívne dobré technické vybavenie škôl (práve vďaka vyššie spomenutým projektom), ktoré sa neustále zlepšuje (viď projekt DiGiŠkola [8]). Len z tohto štruktúrneho projektu bolo do našich škôl dodaných 5680 interaktívnych tabúl, 5680 notebookov, 2686 farebných tlačiarň, 20000 tabletov<sup>1</sup> a 1000 wifi routerov [8].

<sup>1</sup>V slovenských podmienkach zatiaľ nie je odskúšaný model s použitím tabletov/iných mobilných zariadení v bežnej triede (bez nutnosti budovať špeciálnu multimedialnú učebňu)

Ďalšia technika bola školám dodávaná v rámci iných vyššie spomenutých projektov. V súčasnosti majú mnohí učitelia vďaka rôznym národným projektom (Infovek, Modernizácia vzdelávacieho procesu na ZŠ a SŠ, Digitalizácia vzdelávania) aj dostatok vedomostí na tvorbu vlastných elektronických učebných materiálov, avšak pri ich vyťaženi samostatnou výučbou vznikajú tieto materiály len veľmi pomaly. Interaktívne pracovné listy sa používajú len veľmi zriedkavo, keďže sa (mylne) predpokladá možnosť ich nasadenia výlučne v multimediálnej (počítačovej) učebni.

Súčasný stav možno zhrnúť do nasledovných bodov:

- učitelia a žiaci majú dostatok počítačových zručností,
- základné školy majú dostatočné technické vybavenie
- počítač sa na hodine geografie používa len zriedkavo,
- pre vyučovací predmet Geografia pre ZŠ neexistuje komplexný elektronický učebný materiál s príslušnými metodickými pokynmi.

## 2. Základné vlastnosti programu Chamilo

LMS Chamilo je špeciálny interaktívny vzdelávací systém určený na tvorbu, úpravu a správu vzdelávacích projektov, lekcii a elektronických kurzov.

Chamilo vznikol v roku 2010 v dôsledku nespokojnosti viacerých vývojárov s vývojom projektu Dokeos ako nová vývojová vetva systému Dokeos [6]. Ten má svoje korene v systéme Claroline od ktorého sa oddelil v roku 2004 [5]. LMS Chamilo je slobodný (OpenSource) e-learningový softvér šírený pod licenciu GNU GPL3. Výhodou (z pohľadu učiteľa a žiaka ZŠ) je jeho esteticky vkusné grafické spracovanie, nekomplikované používanie a dostupná slovenská lokalizácia.

Učiteľ môže prostredie systému Chamilo zužitkovať predovšetkým v týchto oblastiach:

- zverejnenie základných informácií o predmete — obsahová náplň predmetu,
- hodnotenie, odporúčaná (rozširujúca) literatúra,
- zverejňovanie materiálov pre žiakov/študentov,
- interaktívne pracovné zošity,
- testy pre žiakov,
- komunikácia so žiakmi,
- kooperatívne vyučovanie (tvorba a zdieľanie dokumentov),
- zadania a domáce úlohy.

V súčasnosti (2015) je Chamilo dodávaný v dvoch verziách a to LMS 1.9.10 a LCMS 3.0. Informačný systém je v súčasnosti dostupný aj na portáli Katedry geografie a geológie, Fakulty prírodných vied Univerzity Mateja Bela v Banskej Bystrici, na webovej adrese <http://geo.fpv.umb.sk/chamilo/>, kde aktuálne používame platformu Chamilo LMS 1.9.8 (2015).

## 2.1. Tvorba elektronických učebných materiálov

Práca v prostredí LMS Chamilo nevyžaduje žiadne špeciálne zručnosti, je dostupná bez ohľadu na dosiahnutú úroveň počítačovej gramotnosti a vlastnú aprobáciu. Ako súčasť práce [4] bolo v prostredí Chamilo vytvorených 15 interaktívnych pracovných listov (4–9 položiek) a 4 veľké overujúce interaktívne testy (10–14 položiek) určených na zopakovanie konkrétneho tematického celku. Vypracované elektronické materiály súhrnne pokrývajú približne 85% povinného vzdelávacieho obsahu učiva Geografie pre 5. ročník ZŠ podľa aktuálneho ŠVP [3].

E-learningový softvér Chamilo bol pilotne nasadený do reálneho vyučovacieho procesu na ZŠ Spojová 14 v Banskej Bystrici. Pri práci so systémom Chamilo sme sa venovali tvorbe on-line pracovných listov, testov a učebnej osnovy.

**2.1.1. Interaktívne pracovné listy a testy.** Test alebo pracovný list konštruje autor kurzu (pedagóg) v module s názvom „Testy“ pridaním činnosti „Nový test“, ktorý pomenuje príslušným názvom. Testy si môže učiteľ triediť do kategórií, ktoré si sám určí (napr. podľa ročníka, predmetu alebo zamerania na určitú oblasť geografie). Ku každému testu je možné pripojiť stručný popis, ľubovoľné množstvo otázok, obrázky, fotografie, mapy, náčrty, tabuľky, grafy aj zvuky a rôzne animácie, čím testovanie vedomostí a zručností žiaka nadobúda komplexnejší a objektívnejší rozmer. Systém Chamilo umožňuje autorovi interaktívnych testov (pracovných listov) vyberať z jedenástich kľúčových formátov testových úloh. Každý vytvorenej otázke je potrebné určiť „Váhu“ v podobe bodového hodnotenia aby systém vedel odpovede správne vyhodnotiť a zaškrtnúť správnu (očkávanú) odpoveď.

Rovnako podstatnou úlohou po vytvorení testu je jeho vyhodnotenie. Žiaci po vyplnení interaktívneho testu (pracovného listu) získavajú bleskovú spätnú väzbu ohľadom svojej úspešnosti. Prístup k výsledkom má aj učiteľ, ktorý získava prehľad o tom, ktoré otázky robili žiakom najväčšie problémy a ktoré naopak patrili medzi tie ľahšie, resp. mali najvyššiu úspešnosť.

**2.1.2. Interaktívna učebná osnova.** Účebná osnova predstavuje usporiadanú postupnosť aktivít a výučbových materiálov tak, aby žiak čo najefektívnejšie získal požadované vedomosti a zručnosti (obr. 1). Jednotlivé body osnovy môžu byť prerekvizitami pre ďalšie body, t.j. je možné systém nastaviť tak, že nepustí žiaka na ďalší bod osnovy, bez toho aby nemal splnený bod predchádzajúci.

Tvorba interaktívnej učebnej osnovy v prostredí LMS Chamilo nevyžaduje žiadne špeciálne zručnosti. Vytvára ju autor kurzu (učiteľ) v module s názvom „Učebná osnova“ pridaním činnosti „Pridať novú osnovu“. Vytvorené učebné osnovy si môže učiteľ triediť do kategórií, ktoré si sám určí. Označením ponúkaných hypertextových odkazov vie pedagóg učebnú osnovu upravovať, nastavovať parametre, zviditeľňovať a publikovať ju, exportovať ju na CD/DVD disk alebo

do PDF webovej stránky a vytvárať kópie už existujúcej učebnej osnovy. Pedagóg smie následne vytvorenú interaktívnu učebnú osnovu importovať vo forme SCORM-u, AICC a konvertovať do formátu PPT. Systém Chamilo umožňuje do učebnej osnovy importovať pracovné listy, grafické informácie (obrázky, fotografie, mapy, náčrty, tabuľky, grafy) zvukové záznamy (audio), internetové odkazy, animácie, videá, atď.



Obr. 1. Učebná osnova v systéme Chamilo

Didakticky dobre premyslená učebná osnova môže výrazne ovplyvniť (v pozitívnom, ale aj negatívnom smere) priebeh štúdia, aj samotný vedomostný progres žiaka. Preto je dôkladne premyslený návrh poradia na seba nadväzujúcich učebných činností spolu s odporúčanými postupmi ich riešenia kľúčovým faktorom úspešnosti vzdelávacieho kurzu. Pedagóg môže vytvorenú interaktívnu učebnú osnovu využiť v rôznych fázach vyučovacieho procesu:

1. Expozičná časť vyučovacej hodiny (sprístupňovanie nových poznatkov – možnosť využitia učebnej osnovy ako „interaktívneho pracovného zošita“ pre daný vyučovací predmet).
2. Fixačná časť vyučovacej hodiny (upevňovanie a precvičovanie nadobudnutých vedomostí prostredníctvom interaktívnych testov).
3. Aplikačná časť vyučovacej hodiny (opakovanie získaných poznatkov).

### 3. Pilotné overenie v praxi

Využitelnosť softvéru LMS Chamilo sme testovali v praxi na ZŠ v predmete Geografia pre 5. ročník v šk. roku 2014/2015 v dňoch 20. 11. 2014 a 23. 4. 2015. Pilotné odskúšanie softvéru LMS Chamilo prebiehalo na hodinách informatiky

(nie geografie) a to z dôvodu pripojiteľnosti na internet a potreby počítačovej techniky individuálne pre každého žiaka.

### 3.1. Prvé pilotné testovanie

Prvého pilotného testovania zúčastnilo spolu 24 žiakov. Interaktívny didaktický test obsahoval 10 otázok a bol zameraný na preverenie dlhodobej pamäte žiakov. Systém Chamilo nemenil ich usporiadanie, takže každý žiak riešil rovnaký test. Test nemal presne stanovený časový limit na jeho ukončenie. Priemerný čas, za ktorý vyriešil jeden žiaci vyriešili test bol 12 minút. Počet opakovaní testu nebol obmedzený, čo sa nám aj vypomstilo pri konečnom vyhodnotení, keďže niektorí „šikovní“ žiaci si test zopakovali aj trikrát, čím sme získali skreslené výsledky.

9. Porozmýšľaj a označ, či je dané tvrdenie pravdivé alebo nie.

Nesprávne

Skóre : 4 / 6

| Vaša voľba | Ošakovaná voľba | Odpoveď   | Komentár |
|------------|-----------------|---|----------|
| Pravda     | Nepravda        | Na severnej pologuli je menej pevniny ako na južnej.                  |          |
| Pravda     | Nepravda        | Plocha oceánov na severnej pologuli je väčšia ako plocha pevniny.     |          |
| Pravda     | Pravda          | Povrch Zeme tvorí voda a pevnina.                                     |          |
| Nepravda   | Nepravda        | Pevniny zaberajú viac ako polovicu povrchu Zeme.                      |          |
| Nepravda   | Nepravda        | Na Zemi existujú len tri oceány (Tichý, Atlantický a Severný ľadový). |          |
| Pravda     | Pravda          | Menšiu časť pevniny, ktorá je obklopená vodou, nazývame ostrov.       |          |

Obr. 2. Vzor problémovej dichotomickej testovej úlohy vytvorenej v prostredí Chamilo


### 3.2. Druhé pilotné testovanie

Druhého pilotného testovania zúčastnilo 22 žiakov. Overujúci interaktívny didaktický test obsahoval 10 testových úloh a bol, podobne ako predchádzajúci, zameraný na preverenie dlhodobej pamäte žiakov. V prípade druhého testovania si systém vyberal z 19 ponúkaných testových úloh, ktoré následne ľubovoľne kombinoval, čo znamená, že ani jeden žiak neriešil rovnaký test. Test nemal presne

stanovený časový limit na jeho ukončenie. Priemerný čas, za aký vyriešil jeden žiak nami vytvorený test sme vypočítali na 11 minút. Parameter možnosti opakovaného riešenia testu sme v tomto prípade nastavili len na dva pokusy.

5. Ako nazývame doliny, ktoré majú strmé skalné steny a vznikajú činnosťou rieky (rieka sa zarezáva do tvrdých hornín, napríklad do vápenca)?

Správne Skóre : 1 / 1



| Vaša voľba                       | Očakávaná voľba                  | Odpoveď           |
|----------------------------------|----------------------------------|-------------------|
| <input checked="" type="radio"/> | <input checked="" type="radio"/> | kaňony a tiesňavy |
| <input type="radio"/>            | <input type="radio"/>            | kameňolomy        |
| <input type="radio"/>            | <input type="radio"/>            | úžiny             |
| <input type="radio"/>            | <input type="radio"/>            | prepadiská        |

**Obr. 3.** Ukážka najfrekvencovanejšej (testovú otázku riešilo až 77% žiakov) a najúspešnejšej testovej úlohy s jednou správnou odpoveďou vytvorenej v prostredí Chamilo

## 4. Záver

Evalvácia interaktívnych testov v podmienkach reálneho vyučovacieho procesu a pohotovú spätnú väzbu v systéme Chamilo potvrdila nasledovné vlastnosti využívania e-materiálov vo výučbe:

- časová nenáročnosť prípravy vyučovacích hodín,
- vysoká rýchlosť vyhodnocovania úspešnosti riešenia jednotlivých testových úloh a testovania,
- okamžitá spätná väzba,
- zvýšenie motivácie u žiakov vďaka možnosti obohatenia otázok a úloh o multimediálny obsah (fotografie, videá, zvukové záznamy, texty, animácie, grafy, tabuľky, atď.) do jednotlivých testových úloh,
- jednoduché ovládanie,
- grafická zrozumiteľnosť a prehľadnosť hypertextových odkazov,
- rýchla a pohodlná tvorba elektronických učebných materiálov.

Významným nedostatkom je dlhodobo neriešená chybné spracovanie ukladania resp. čítania z databázy pri ktorom sa nepoužíva jednotne kódovanie UTF8, čoho dôsledkom je nesprávne vyhodnocovanie doplnovacích úloh obsahujúcich Non-latin1 znaky. Aj v prípade správnej odpovede ju systém vyhodnotí ako nesprávnu. Všetkými ostatnými vlastnosťami prezentovaný e-learningový systém

spĺňa všetky požiadavky, ktoré naň boli z hľadiska využitia v edukačnom procese geografie kladené a po dopracovaní materiálov z iných ročníkov by mohol plniť funkciu digitálnej cvičebnice/učebnice z geografie.

## Literatúra

- [1] BALÁŽOVIČ, L.: *Claroline – jednoduchý e-learning v praxi*, Otvorený softvér vo vzdelávaní, výskume a IT riešeníach, zborník medzinárodnej konferencie OSSConf 2014, Žilina, 2.–4. júla 2014, str. 33–40, ISBN 978-80-970457-4-6, <http://ossconf.soit.sk/images/zborniky/zbornik2014.pdf>.
- [2] KALIŇÁK, M.: *Planéta vedomostí sa dostáva do škôl aj domácností*, <https://www.minedu.sk/planeta-vedomosti-sa-dostava-do-skol-aj-domacnosti/>, tlačová správa MŠVVŠ SR, [cit. 23. 3. 2015].
- [3] NOGOVÁ, M.: *Štátny vzdelávací program Geografia (Vzdelávacia oblasť: Človek a spoločnosť) PRÍLOHA ISCED 2*, ŠPÚ Bratislava, 2010, [http://www.statpedu.sk/files/documents/svp/2stzs/isced2/vzdelavacie\\_oblasti/geografia\\_isced2.pdf](http://www.statpedu.sk/files/documents/svp/2stzs/isced2/vzdelavacie_oblasti/geografia_isced2.pdf), [cit. 23. 3. 2015].
- [4] PILIŠ, V.: *Tvorba a pilotné overenie elektronických učebných materiálov v geografii na ZŠ*, diplomová práca, UMB, Banská Bystrica, 2015, 91 s.
- [5] Dokeos – LIMSWiki, <http://www.limswiki.org/index.php?title=Dokeos>, [cit. 8. 6. 2015].
- [6] Chamilo, <http://en.wikipedia.org/w/index.php?title=Chamilo&oldid=656473847>, [cit. 8. 6. 2015].
- [7] Moderné vzdelávanie – digitálne vzdelávanie pre všeobecno-vzdelávacie predmety, CVTI SR, Bratislava 2013, [cit. 23. 3. 2015], <http://www.digitalnevzdelavanie.sk/o-projekte/modernizacia-skolstva>.
- [8] O projekte Digiškola, <http://new2.digiskola.sk/o-projekte/>, [cit. 8. 6. 2015].
- [9] Projekt Infovek, Asociácia projektu Infovek, 2009–2014, <http://www.infovek.sk/>, [cit. 23. 3. 2015].
- [10] Projekt Modernizácia vzdelávania na základných a stredných školách, Elfa, Košice 2010, <http://www.modernizaciavzdelavania.sk>, [cit. 23. 3. 2015].
- [11] Zborovna.sk, Komunitný portál, Komenský s. r. o., 2008–2015, <http://www.zborovna.sk>, [cit. 23. 3. 2015].

## Kontaktné adresy

**Mgr. Vladimír Piliš**, Katedra geografie, geológie a krajinnej ekológie, Univerzita Mateja Bela, Tajovského 40, 974 01 Banská Bystrica, Slovenská Republika,  
*E-mailová adresa:* [vladopilis@gmail.com](mailto:vladopilis@gmail.com)

**Mgr. Ľuboš Balážovič, PhD.**, Katedra geografie, geológie a krajinnej ekológie, Univerzita Mateja Bela, Tajovského 40, 974 01 Banská Bystrica, Slovenská Republika,  
*E-mailová adresa:* [lubos.balazovic@umb.sk](mailto:lubos.balazovic@umb.sk), <http://www.fpv.umb.sk/lbalazovic/>



## NÁVRHÁŘ STYLŮ V APLIKACI T<sub>E</sub>XonWeb

JAN PŘICHYSTAL (CZ)

**Abstrakt.** Efektivní tvorba dokumentů vytvářených pomocí počítače je založena na principu strukturálního značkování. V krátkosti jde o to, určit si v dokumentu jednotlivé strukturální celky (nadpisy, tabulky, obrázky, . . . ), vytvořit pro ně adekvátní značky či styly a následně definovat jejich vzhled. Cílem je oddělit vizuální a obsahovou stránku dokumentu tak, aby bylo možné při tvorbě dalšího obdobného dokumentu využít již existující šablonu nebo naopak vytvořit pro jeden dokument vícero různých vzhledů podle aktuální potřeby a to vše bez nutnosti zasahovat do vlastního obsahu dokumentu. Strukturální značkování je neoddelitelně spjata i s tvorbou dokumentů v systému T<sub>E</sub>X. Jeho nadstavba L<sup>A</sup>T<sub>E</sub>X se snaží nabízet množinu již definovaných stylů, které může uživatel používat pro formátování různých typů dokumentů. Nicméně se ukazuje, že vždy se najde situace, kdy předdefinované styly nestačí požadavkům autora a musí se komplikovaně modifikovat. Další možností je tedy vytvoření vlastních stylů. S tím však mají začínající uživatelé, kteří T<sub>E</sub>X používají, problémy. Této situaci se snaží čelit webová aplikace T<sub>E</sub>XonWeb, která nabízí podobný přístup při tvorbě stylů, jaký uživatelé důvěrně znají z jiné aplikace pro tvorbu dokumentů, z MS Word.

**Klíčová slova.** T<sub>E</sub>XonWeb, T<sub>E</sub>X/L<sup>A</sup>T<sub>E</sub>X, webová aplikace, návrhář stylů.

### STYLE DEFINITION IN T<sub>E</sub>XONWEB APPLICATION

**Abstract.** Efficient creation of documents created by computer is based on the principle of structural markup. In short, the point is to identify the individual structural units of the document (headings, tables, pictures, . . . ), create for them proper macros or styles and then define their visual interpretation. The aim is to separate the visual side and content of the document so that it is possible to use an existing template in the creation of another similar document, or vice versa to create several different looks for one document according to current needs and all without having to interfere with the content of the document. Structural markup is also inextricably linked with creating documents in T<sub>E</sub>X system. The L<sup>A</sup>T<sub>E</sub>X is trying to offer a set of predefined styles, the user can use for formatting of various types of documents. However, it appears that there are always situations where the predefined styles are insufficient for author requirements and must be modified with difficulty. Another option here is to create own user styles. With that, however, novice users using T<sub>E</sub>X have problems. This situation is trying to face T<sub>E</sub>XonWeb web application that offers a similar approach when creating styles which users are familiar with from other applications for document preparation, e. g. from MS Word.

**Keywords.** T<sub>E</sub>XonWeb, T<sub>E</sub>X/L<sup>A</sup>T<sub>E</sub>X, web application, style designer.

## Úvod

Webová aplikace T<sub>E</sub>XonWeb se již několik let snaží zjednodušit a zpříjemnit začínajícím uživatelům seznámení se a používání typografického systému T<sub>E</sub>X. Pro tento účel aplikace nabízí různé funkce, které uživatelům usnadňují zápis zdrojového kódu. Jde například o zvýrazňování syntaxe, zobrazování párových závorek, tlačítka pro vkládání často používaných značek či symbolů, šablony dokumentů či průvodce tvorbou komplikovanějších T<sub>E</sub>Xových struktur. Mezi ně patří návrhář seznamů a tabulek a nástroj pro vkládání obrázků. Podrobnější popis motivace vzniku aplikace T<sub>E</sub>XonWeb uvádí například [2].

Jak je patrné při výuce předmětu Zpracování textů na počítači, který studuje pravidelně každý semestr přibližně tři sta studentů Mendelovy univerzity, právě návrháři zdrojového kódu (wizardi) jsou studenty často využíváni a usnadňují jim nejen zápis kódu, ale i pochopení toho, jak se konkrétní část dokumentu v systému T<sub>E</sub>X zapisuje. Z tohoto důvodu jsme se rozhodli pokusit se vytvořit i návrhář stylů, tak jak je znám například z běžných textových procesorů, který by začínajícím uživatelům umožnil tvorbu vlastních značek pro strukturní označování dokumentu a usnadnil jim seznámení s problematikou definice vlastních maker.

## Současný stav

V současné době je aplikace T<sub>E</sub>XonWeb využívána především studenty Mendelovy univerzity při výuce předmětu Zpracování textů na počítači. Tuto aplikaci využívají také pro řešení úkolů a závěrečného projektu, pro tvorbu svých závěrečných bakalářských a magisterských prací. Dále je T<sub>E</sub>XonWeb využíván i některými akademickými pracovníky pro tvorbu posudků závěrečných prací. To vše umožňují nabízené šablony zjednodušující tvorbu standardních dokumentů. Mezi dalšími k dispozici například šablona pro tvorbu životopisů, prezentací, žádostí, dopisů a dalších.

Z toho je patrné, že typickým uživatelem T<sub>E</sub>XonWeb je opravdu začínající či běžný uživatel T<sub>E</sub>Xu, který většinou využívá nadstavbu L<sup>A</sup>T<sub>E</sub>X a předdefinované šablony. Nicméně, právě při výuce předmětu Zpracování textů na počítači se studenti seznamují s problematikou strukturního značkování. Je jim na příkladu MS Word demonstrováno striktní používání vlastních či předdefinovaných stylů a výhody tohoto přístupu. Stejným způsobem je tato problematika probírána i na příkladu T<sub>E</sub>Xu respektive L<sup>A</sup>T<sub>E</sub>Xu. Studenti si z internetu stáhnou vybraný článek, identifikují v něm jednotlivé strukturní celky a nadefinují pro ně odpovídající makra, kterými následně článek označují. Na tomto příkladě vidí výhody tohoto přístupu, který jim umožňuje definovanou šablonu znovu použít pro jiný dokument a dosáhnout tak stejného vzhledu, případně použít různé šablony pro jeden dokument a mít tak více tvarů jednoho textu. Nedílným krokem tohoto postupu

je definice vlastních značek pomocí makra `\def`. To je pro začínající uživatele, považmo pro studenty neinformatických oborů, poněkud komplikované a mají s tím problémy. Naopak při definici vlastních stylů v MS Word problémy nejsou. To nás vedlo k nápadu přiblížit studentům tuto problematiku podobným způsobem, jako při tvorbě tabulek či seznamů, pomocí návrháře, který by svým vzhledem a funkcemi připomínal tvorbu stylů v MS Word.

V současné chvíli aplikace T<sub>E</sub>XonWeb žádného takového návrháře nenabízí a uživatelé musí vlastní značky definovat ručně. Jak je u začínajících uživatelů běžné, dopouští se spousty chyb, což při definici vlastních maker často způsobí, že dokument ani nelze přeložit. Oblíbenou chybou je totiž například zapomenutá závorka na konci definice. Návrhář stylů by mohl tyto problémy pomocí minimalizovat a sloužit jako jakýsi návod, jak vlastně při tvorbě vlastních značek postupovat.

## Analýza problému

Jak už bylo uvedeno výše, cílem je vytvořit návrháře kódu tak, aby svým vzhledem a funkcemi připomínal tvorbu stylů z MS Word. Je zde však několik limitujících faktorů, které řešení komplikují. Jedná se především o to, že vytvořený kód by mělo být možné nejen vložit do dokumentu, ale také znovu otevřít v návrháři a pokračovat v jeho editaci. Dalším problémem je rozdílný charakter programů MS Word a T<sub>E</sub>X, který způsobuje, že některá nastavení vlastností v MS Word nemají v T<sub>E</sub>Xu smysl a naopak. Dalším limitujícím faktorem je i to, že T<sub>E</sub>XonWeb je určen i pro použití na dotykových zařízeních [1] což komplikuje tvar návrháře. Z těchto i dalších důvodů je nutné řešení přizpůsobit daným podmínkám a upravit jej tak, aby vyhovovalo použití v konkrétní aplikaci.

Jedním z prvních kroků při analýze problému je tedy prostudování atributů, které tvorba stylů v MS Word nabízí a výběr vhodných kandidátů, kteří by se mohli v návrháři objevit. Dále je nutné rozmyslet, jak bude návrhář v T<sub>E</sub>XonWeb vypadat a jak bude realizován, aby byl použitelný například právě i na dotykových zařízeních. Následně bude nutné promyslet, jakým způsobem bude realizováno vkládání vytvořeného kódu definovaných maker a jejich načítání již vloženého kódu zpět do návrháře.

## Návrh řešení

Na základě problémů identifikovaných v předchozí části jsme vytipovali množinu vhodných atributů, které budou součástí návrháře stylů v aplikaci T<sub>E</sub>XonWeb. Tato množina rozhodně nepokrývá všechny vlastnosti částí dokumentů, které je možné modifikovat. To ani není jejím cílem. Při výběru jsme vycházeli především z množiny znalostí vyučovaných v předmětu Zpracování textů na počítači,

konkrétně v kapitolách Smíšená sazba a Odstavcová sazba. Tato množina atributů byla dále zmenšena na atributy, které běžně používáme při tvorbě vlastních strukturních značek při výuce tohoto předmětu. Jde o to nabídnout v podstatě výukový nástroj formou ukázek nikoliv komplexní řešení. Výsledný přehled je obsahem následující tabulky.

| <b>Písmo</b>           |   |
|------------------------|---|
| <b>atribut</b>         | <b>zápis</b>  |
| rodina písma           | <code>\fontspec</code>  |
| neproporcionální písmo | <code>\texttt</code>  |
| bezpatkové písmo       | <code>\textsf</code>  |
| tučný řez písma        | <code>\textbf</code>  |
| skloněný řez písma     | <code>\textsl</code>  |
| kurzívní řez písma     | <code>\textit</code>  |
| kapitálky              | <code>\textsc</code>  |
| velikost písma         | <code>\fontsize</code> resp. <code>\tiny</code> až <code>\Huge</code> |
| barva písma            | <code>\color</code>   |

| <b>Odstavec</b>        |   |
|------------------------|---|
| zarovnání vlevo        | <code>\raggedright</code>   |
| zarovnání vpravo       | <code>\raggedleft</code>  |
| zarovnání na střed     | <code>\centering</code>   |
| odstavcová zarážka     | <code>\parindent</code>   |
| vertikální odsazení    | <code>\parskip</code>   |
| mezera východové řádky | <code>\parfillskip</code>   |
| levý okraj             | <code>\leftskip</code>  |
| pravý okraj            | <code>\rightskip</code>   |
| řádkování              | <code>\baselineskip</code>  |
| svázat s následujícím  | <code>\nolinebreak</code>   |
| vertikální mezera      | <code>\hspace</code> resp. <code>\smallskip</code> až <code>\bigskip</code> |
| poznámka pod čarou     | <code>\footnote</code>  |
| křížové odkazy         | <code>\label</code> a <code>\ref</code>                                     |

Tyto atributy budou rozděleny do jednotlivých karet po skupinách, jak je patrné z uvedené tabulky. Tyto atributy umožní definovat v podstatě jen znakové styly,

tak jak jsou známy z MS Word. Komplexnější nastavení dokumentu zatím není v návrháři stylů plánováno.

Postup definice vlastní strukturní značky bude podobný tomu, jak je řešen v MS Word. Uživatel určí zda chce vytvářet nový styl nebo modifikovat již existující. Dále určí název značky (stylu) nadefinuje hodnoty požadovaných atributů. Jednotlivé atributy budou defaultně obsahovat předdefinované hodnoty, které jsou běžně pro danou třídu dokumentu nastaveny. Všechny vytvořené styly budou dostupné v přehledovém okně a uživatel bude moci výběrem konkrétního stylu editovat jeho obsah.

Otázkou je, jakým způsobem zajistit, že změny, které uživatel pomocí návrháře provede, bude možné po jejich vložení do dokumentu později opět v návrháři modifikovat. Problém je především v tom, že možných úprav definic, které uživatel ručně v kódu může provést, je neomezené množství a parsovat T<sub>E</sub>Xový kód není v silách jednoduchého návrháře. Kód, který vznikne z návrháře tedy nebude možné uživatelsky upravovat. Pokud k tomu dojde, nebude možné jej nadále upravovat pomocí návrháře a uživatel bude muset kód dále upravovat pouze ručně. Optimální pro tento postup by tedy bylo, kdyby část kódu s definicemi byla v editoru dostupná pouze v režimu pro čtení. To však Ace editor v současné době neumožňuje. Blíže se možnostem editoru integrovaného do aplikace T<sub>E</sub>XonWeb věnuje [3].

Řešením, které by mohlo zajistit needitovatelnost kódu definic v použitém editoru, je například oddělení definic do samostatné části dokumentu ohraničené komentářem. Jeho součástí bude hash daného kódu. Pokud dojde k redefinici kódu, uvedený hash se nebude shodovat s hashem aktuálně vypočteným a editace nebude v návrháři umožněna. Jde o neoptimální řešení, ale v současné situaci nejsme schopni jednoduše zajistit celistvost návrháře jinou cestou.

Co se týče použitelnosti návrháře na dotykových zařízeních, je nutné provést podrobnou analýzu toho, jak umožnit efektivně zadávat a modifikovat tak velké množství hodnot, aby se v nich uživatel neztratil a následně implementované řešení intenzivně otestovat. K tomu hodláme využít studenty, kteří budou T<sub>E</sub>XonWeb používat ve výuce předmětu Zpracování textů na počítači v zimním semestru 2015/2016. Výsledky hodláme publikovat na konferenci PEFNet či Firma a konkurenční prostředí, které se konají na Mendelově univerzitě v Brně.

## Závěr

Tento článek popisuje návrh řešení návrháře stylů v rámci webové aplikace T<sub>E</sub>XonWeb. Věnuje se identifikaci problémových míst při analýze problému, vybírá seznam atributů, které budou součástí tohoto návrháře a popisuje řešení problémů, které při implementaci a používání mohou nastat.

Pokud bychom se chtěli inspirovat při hledání řešení u konkurenčních projektů jako jsou shareL<sup>A</sup>T<sub>E</sub>Xa Overleaf, neuspějeme. V současné době tyto dva projekty,

kteře nabízí podobnou funkcionalitu jako T<sub>E</sub>XonWeb, návrhář stylů neobsahuje. Po implementaci takového pomocníka tvorby komplikovaných částí zdrojového textu systému T<sub>E</sub>X bude mít aplikace T<sub>E</sub>XonWeb výhodu a může uživatelům přinést něco navíc.

## Reference

- [1] PŘICHYSTAL, J.: *Možnosti tvorby dokumentů v T<sub>E</sub>Xu pomocí webového prohlížeče*, Otvorený softvér vo vzdelávaní, výskume a v IT riešeniach, zborník medzinárodnej konferencie OSSConf 2014, Žilina, 2.–4. júla 2014, str. 23–30, ISBN 978-80-970457-4-6, <http://ossconf.soit.sk/images/zborniky/zbornik2014.pdf>.
- [2] PŘICHYSTAL, J. RYBIČKA, J.: *Webové rozhraní pro sazbu dokumentů*, Zpravodaj CSTUG, 2004, sv. 14, č. 3, s. 190–194, ISSN 1211-6661.
- [3] VYBÍHAL, J. PŘICHYSTAL, J.: *Tvorba nového uživatelského rozhraní aplikace T<sub>E</sub>XonWeb*, PEFnet 2014, 1. vyd., Brno: Mendelova Univerzita v Brně, 2014, s. 1–10, ISBN 978-80-7509-152-9.

## Kontaktní adresa

**Ing. Jan Přichystal, Ph.D.**, Provozně ekonomická fakulta, Mendelova univerzita v Brně,  
Zemědělská 1, 613 00 Brno, Česká Republika,

*E-mailová adresa:* [jan.prichystal@mendelu.cz](mailto:jan.prichystal@mendelu.cz)

## ROUTING V POSTGIS NAD DATY SILNIČNÍ DATABANKY

JAN RŮŽIČKA (CZ)

**Abstrakt.** Příspěvek si klade za cíl otevřít diskusi na téma možností dat silniční databanky pro hledání nejkratších (nejrychlejších) spojení v silniční síti. Pro účely projektu, který se věnuje dopravním informacím byly posuzovány možnosti využití dat silniční databanky pro hledání nejkratších tras v silniční síti. Hlavním důvodem posouzení byla možnost napojení na informace ze sčítání dopravy. Při realizaci tohoto posouzení se vycházelo z několika základních požadavků, kde jedním z nich bylo využití zpracování na serveru s využitím open source nástrojů. Jako vhodný nástroj se jevil systém PostgreSQL/PostGIS s rozšířením pgRouting. Vyhledávání tras se podařilo realizovat poměrně bez zásadních komplikací, přesto vyvstalo několik zásadních otázek, které by měly být zodpovězeny.

**Klíčová slova.** Routing, PostGIS, geodata, licence.

## ROUTING IN POSTGIS WITH DATA FROM ROAD NETWORK OF THE CZECH REPUBLIC

**Abstract.** The paper is focused on starting a discussion about possibilities of geodata from Road Network of the Czech Republic for searching shortest (fastest) connections in the network. The possibilities of the geodata were tested for purposes of project that is focused on traffic information analysis. The main reason for the evaluation was a possibility of connection to data about statistics from traffic census. The evaluation was based on several basic requirements, where the main one was usage of open source technologies. System PostgreSQL/PostGIS with pgRouting extension was selected. Routing was done without any difficult problems, but results produced several questions that should be discussed.

**Keywords.** Routing, PostGIS, geodata, licence.

## Úvod

Pokud potřebujeme hledat nejkratší spojení v dopravní síti můžeme k tomu využít nástroje geografických informačních systémů. Řada desktop nástrojů jimi disponuje a je to tedy poměrně triviální úloha. Pokud však máme data dané sítě, případně další data pro následující analýzy uloženy v relační databázi, může se stahování dat na stranu klienta jevit jako nevýhodné. Také pokud potřebujeme provádět dané výpočty např. ve vazbě na webové řešení, pak využití desktop klienta není úplně to nejlepší řešení. Naštěstí jsou již k dispozici serverová řešení a dokonce taková, která běží jako součást databáze. Výpočty je tedy možno

realizovat přímo v prostředí relační databáze a to s využitím jazyka SQL. Vyhledávání tras se podařilo realizovat poměrně bez zásadních komplikací, přesto vyvstalo několik zásadních otázek, které by měly být zodpovězeny. Hlavním problémem, který by měl vést k odborné diskusi je otázka přístupu k datům silniční databanky. Data silniční databanky můžete pro účely výzkumu získat z ředitelství silnic a dálnic ČR. Informace o sčítání dopravy můžete získat z www stránek ředitelství. Bohužel takto získaná data jde obtížně propojit, protože systém identifikace úseků se v uvedených datových souborech liší. Údaje ze sčítání dopravy je možné rovněž prohlížet s využitím WMS služby, kde je již identifikace úseků stejná jako u dat použitých pro hledání nejkratších cest. Bohužel WMS služba poskytuje tyto údaje pouze na základě operace GetFeatureInfo, tedy vždy jen v omezené oblasti (pro zjednodušení řekněme: jeden dotaz – jeden úsek). Je etické takto (tedy zasláním většího množství dotazů) získat údaje z WMS služby, pro účely propojení obou databází? Věřím, že tento příspěvek pomůže zahájit tuto diskusi. V rámci příspěvku budou rovněž diskutovány další otázky, které mohou být zajímavé a byly podrobeny analýze, včetně možností propojení dat databáze Open Street Map na data silniční databanky.

## 1. Nástroje

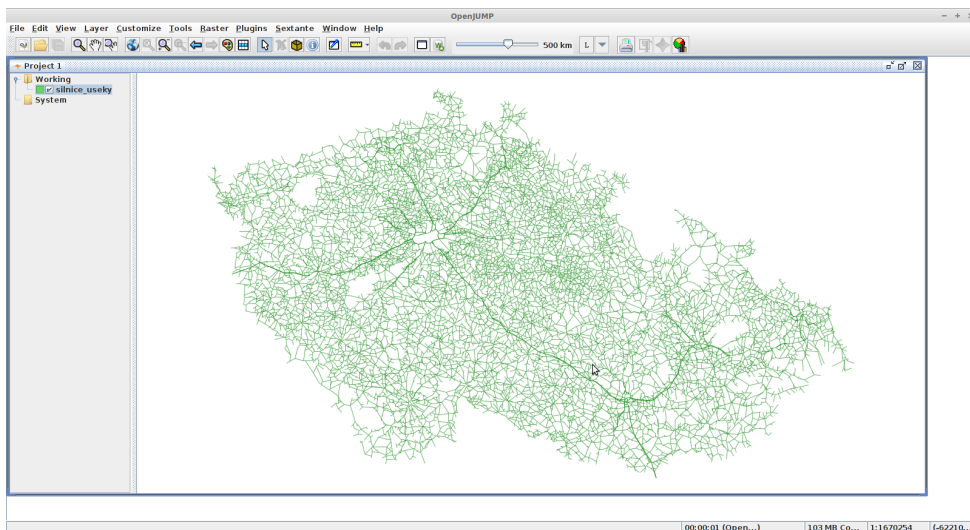
Cílem projektu bylo posouzení vhodnosti geodat silniční databanky pro hledání spojení mezi základními sídelními jednotkami. Základní sídelní jednotky jsou části obcí (nebo celé obce v případě menších obcí) definované na základě pravidel specifikovaných Českým statistickým úřadem. Vyhledání nejkratších spojení je možné realizovat s využitím různých dat o silniční síti (např. volně dostupných dat Open Street Map [2]). Data silniční databanky byla posuzována z důvodu vazby na sčítání dopravy, kde se uvažovalo o využití tohoto sčítání pro další analýzy.

Data silniční databanky v podobě zjednodušených úseků jsou v neaktuální podobě k dispozici např. jako součást kolekce FreeGeodataCZ [1]. Jedná se o spojnice mezi křižovatkami se sadou atributů, jako je např. délka úseku nebo třída silnice. Vše demonstruje obrázek 1.

## 2. pgRouting

Systém řízení báze dat, ve kterém byla geodata uložena je PostgreSQL. Práci s prostorovými daty zajišťuje extenze nazvaná PostGIS [4]. Díky této extenzi je možné pomocí SQL dotazů realizovat prostorové dotazy, např. výběr všech obcí, kterými prochází dálnice D1. Následující obrázek demonstruje uživatelské rozhraní nástroje pgAdmin, který umožňuje práci se systémem PostgreSQL. Na obrázku 2 je ukázka dat z registru RÚIAN (Registr územní identifikace adres a nemovitostí), konkrétně body reprezentující budovy.





**Obrázek 1.** Zjednodušené úseky (Zdroj: FreeGeodataCZ (Ředitelství silnic a dálnic – silniční databanka))

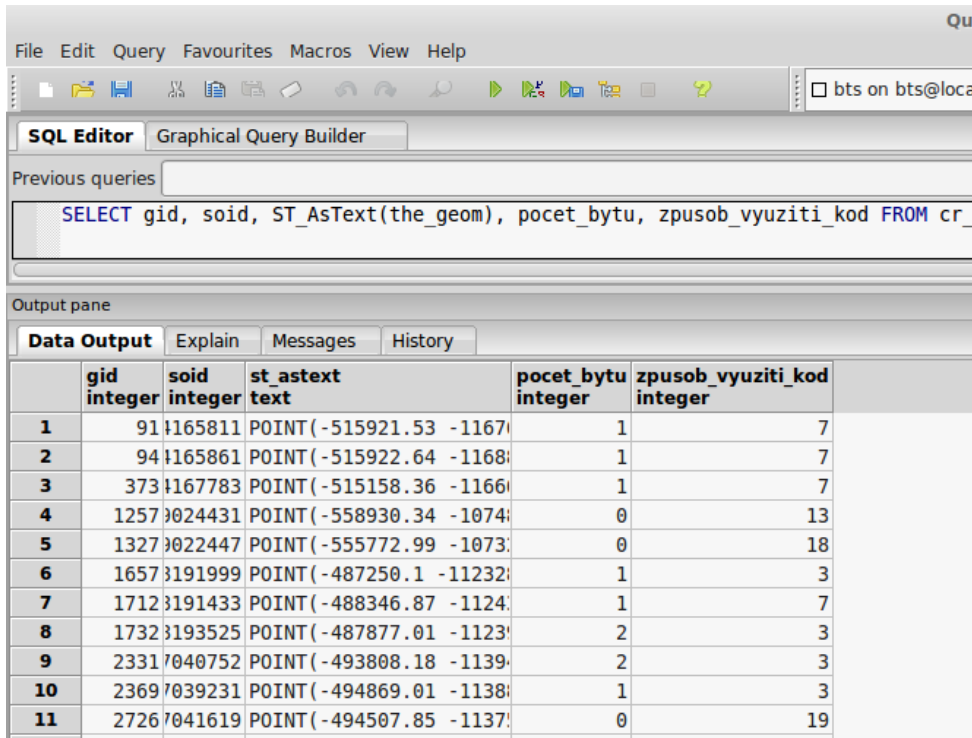
pgRouting [3] je rozšíření PostGIS pro síťové analýzy. K dispozici je celá řada algoritmů, včetně algoritmů využívajících heuristiky. Pro vyhledávání nejkratších tras byl v případě tohoto projektu využit Dijkstrův algoritmus. Použití nástroje pgRouting je poměrně jednoduché, pokud je dopravní síť připravena korektně. V případě Open Street Map je k dispozici nástroj, který dokáže data transformovat k využití v rámci pgRouting. V případě geodat silniční databanky je využití také poměrně jednoduché. Stačí využít funkci `pgr_createTopology`, která zajistí vytvoření doplňujících informací pro vybudování grafu nad sítí. Toto demonstruje následující ukázka kódu v jazyce SQL.

```
ALTER TABLE rsd_cr_useky ADD COLUMN "source" integer ;
ALTER TABLE rsd_cr_useky ADD COLUMN "target" integer ;
```

```
SELECT pgr_createTopology('rsd_cr_useky', 0.00001, 'geom', 'gid');
```

Vyhledání cesty je pak realizováno s využitím SQL dotazu, kde se specifikuje počáteční uzel a koncový uzel, tak jak demonstruje následující příklad v jazyce SQL.

```
SELECT seq, route.id1 AS node, route.id2 AS edge, route.cost,
useky.geom FROM pgr_dijkstra('SELECT gid AS id, source::
integer, target::integer, delka_us::double precision AS cost
FROM rsd_cr_useky', 30, 60, false, false) AS route,
rsd_cr_useky AS useky WHERE route.id2 = useky.gid;
```



The screenshot shows a PostgreSQL/PostGIS SQL Editor window. The query editor contains the following SQL query:

```
SELECT gid, soid, ST_AsText(the_geom), pocet_bytu, zpusob_vyuziti_kod FROM cr_;
```

The output pane displays the results in a table with the following columns: gid (integer), soid (integer), st\_astext (text), pocet\_bytu (integer), and zpusob\_vyuziti\_kod (integer). The results are as follows:

|    | gid<br>integer | soid<br>integer | st_astext<br>text       | pocet_bytu<br>integer | zpusob_vyuziti_kod<br>integer |
|----|----------------|-----------------|-------------------------|-----------------------|-------------------------------|
| 1  | 91             | 165811          | POINT(-515921.53 -1167) | 1                     | 7                             |
| 2  | 94             | 165861          | POINT(-515922.64 -1168) | 1                     | 7                             |
| 3  | 373            | 167783          | POINT(-515158.36 -1166) | 1                     | 7                             |
| 4  | 1257           | 2024431         | POINT(-558930.34 -1074) | 0                     | 13                            |
| 5  | 1327           | 2022447         | POINT(-555772.99 -1073) | 0                     | 18                            |
| 6  | 1657           | 191999          | POINT(-487250.1 -11232) | 1                     | 3                             |
| 7  | 1712           | 191433          | POINT(-488346.87 -1124) | 1                     | 7                             |
| 8  | 1732           | 193525          | POINT(-487877.01 -1123) | 2                     | 3                             |
| 9  | 2331           | 2040752         | POINT(-493808.18 -1139) | 2                     | 3                             |
| 10 | 2369           | 2039231         | POINT(-494869.01 -1138) | 1                     | 3                             |
| 11 | 2726           | 2041619         | POINT(-494507.85 -1137) | 0                     | 19                            |

Obrázek 2. Práce s PostgreSQL/PostGIS

### 3. Využití výsledků hledání cest

Vyhledání spojení mezi základními sídelními jednotkami (ZSJ) proběhlo bez problémů, následně tedy bylo zjišťováno jakým způsobem je možné dané výsledky využívat.

Licence, pod kterou mi byla data poskytnuta je následující: „Ředitelství silnic a dálnic ČR (dodavatel) poskytuje uživateli (Jan Růžička) datové vrstvy zjednodušených uzlů a úseků uzlového lokalizačního systému (ULS) ve formátu ESRI shp (JTSC). Data zůstávají v plném rozsahu v majetku a správě Ředitelství silnic a dálnic ČR. Uživatel není oprávněn data dále poskytnout ani šířit, a to ani v zpracovaném tvaru. Při použití dat uvádějte jako zdroj dat Ředitelství silnic a dálnic ČR.“

Dle poskytnuté licence je tedy jistě možné dále distribuovat seznam ZSJ a přehled ZSJ přes, které se nejkratší cestou dostanu z jedné ZSJ do druhé. Další využití již tak jistě dle licence není. Mohu např.:

1. Dále distribuovat i identifikátory úseků, kterými daná trasa prochází?
2. Vytvořit a dále distribuovat mapu, pokud uvedu zdroj?

3. Distribuovat geometrii úseků z OSM, které namapují na úseky silniční databanky?
4. Distribuovat geometrii úseků z FreeGeoDataCZ?

#### 4. Vazba na sčítání dopravy

Sčítání dopravy je k dispozici v podobě XLS souborů ke stažení (obrázek 3) na stránkách ŘSD nebo v podobě WMS (Web Map Service) služby [5].

|    | A  | B      | C      | D      | E          | F | G                 | H                 |
|----|----|--------|--------|--------|------------|---|-------------------|-------------------|
| 1  | D1 | 1-8024 | 18 610 | 69 414 | 436 88 460 |   | <u>Spořilov</u>   | <u>Chodov</u>     |
| 2  | D1 | 1-8025 | 17 141 | 61 859 | 360 79 360 |   | <u>Chodov</u>     | <u>Průhonice</u>  |
| 3  | D1 | 1-8026 | 17 193 | 51 210 | 187 68 590 |   | <u>Průhonice</u>  | <u>Modletice</u>  |
| 4  | D1 | 1-8023 | 16 184 | 54 163 | 194 70 541 |   | <u>Modletice</u>  | <u>Říčany</u>     |
| 5  | D1 | 1-8027 | 15 180 | 58 020 | 197 73 397 |   | <u>Říčany</u>     | <u>Všechromy</u>  |
| 6  | D1 | 1-8028 | 13 570 | 50 027 | 175 63 772 |   | <u>Všechromy</u>  | <u>Mirošovice</u> |
| 7  | D1 | 1-8030 | 10 226 | 30 762 | 118 41 106 |   | <u>Mirošovice</u> | <u>Hvězdonice</u> |
| 8  | D1 | 1-8040 | 10 128 | 30 183 | 115 40 426 |   | <u>Hvězdonice</u> | <u>Ostředek</u>   |
| 9  | D1 | 1-8050 | 10 045 | 29 072 | 114 39 231 |   | <u>Ostředek</u>   | <u>Štemov</u>     |
| 10 | D1 | 1-8060 | 9 688  | 27 259 | 107 37 054 |   | <u>Štemov</u>     | <u>Psáře</u>      |
| 11 | D1 | 1-8070 | 8 770  | 27 437 | 89 36 296  |   | <u>Psáře</u>      | <u>Soutice</u>    |
| 12 | D1 | 1-8080 | 9 480  | 25 957 | 101 35 538 |   | <u>Soutice</u>    | <u>Loket</u>      |
| 13 | D1 | 1-8089 | 9 280  | 24 563 | 97 33 940  |   | <u>Loket</u>      | <u>Hořice</u>     |

Obrázek 3. Sčítání dopravy v XLS

Výsledky dostupné v podobě XLS souborů však není možné přímo namapovat na zjednodušené úseky, protože identifikátory úseků si neodpovídají. Zůstává tedy možnost využít operaci GetFeatureInfo služby WMS, kde je možné se dostat k údajům v podobě, kterou již je možné na úseky namapovat. Ukázka následuje.

[http://geoportal.jsdi.cz/ArcGIS/services/geoportal\\_rsd\\_wms2/MapServer/WSServer?](http://geoportal.jsdi.cz/ArcGIS/services/geoportal_rsd_wms2/MapServer/WSServer?)

REQUEST=GetFeatureInfo&SERVICE=WMS&LAYERS=0&

CRS=EPSG:5514&WIDTH=512&HEIGHT=512&VERSION=1.3.0&

FORMAT=image/png&STYLES=&

QUERY\_LAYERS=0&X=256&Y=256&INFO\_FORMAT=text/plain&

BBOX=-779883.6875,-1110078.5,-774883.6875,-1105078.5

@0 OBJECTID;Shape;CIS\_SU;ADMINJ;DELKA\_SU;LN;NSN;SN;SNP;

TN;TNP;A;AK;TR;TRP;TV;O;M;SV;TNV;ALFA;BETA;CYKLISTE;

POMER\_SM1;POMER\_SM2;SILNICE;KOD\_TR\_KOM;ZU;KU;POC\_USNASU;

```
KON_USNASU;SUKONVUZLU;ST_ZACSUUS;ST_KONSUUS;C_KOM;STANOVISTE;  
VYM_TAHY;K1;K2;PORADI_US;ADM_OSTAT;
```

```
1272;Polyline;2-0210;CZ0314;  
5585;781;476;204;166;40;164;105;0;9;8;1953;7673;53;9679;  
2145;0;1,28;4;0;0;4;2;x s 19;x s 1758 od Rakovic;  
2223A001 2223A023;2223A012 2223A013;  
K;0;346;4;Krsice;;1;1;126;;
```

Data pro celou ČR tedy byla stažena a namapována na zjednodušené úseky.

## 5. Využití dat ze sčítání dopravy

V dokumentaci k WMS je uvedeno: Fees:zadne, AccessConstraints:bez omezení.

Mohu tedy např.

1. Dále distribuovat Základní sídelní jednotky (spojení mezi nimi) a počet vozidel ze sčítání dopravy?
2. Vytvořit a dále distribuovat mapu intenzity dopravy s využitím Open Street Map nebo úseků z FreeGeoDataCZ?
3. Vytvořit a dále distribuovat mapu srovnávající nejkratší (nejrychlejší) cesty s intenzitou dopravy s využitím Open Street Map nebo úseků z FreeGeoDataCZ?

## 6. Diskuse

Vytvořil jsem tedy dopis, který shrnuje tyto otázky a ten zaslal odborníkům na autorské právo a také na ŘSD. Odpověď z ŘSD mě naprosto odzbrojila, protože jsem dostal ANO na všechny mé otázky.

Jeden z odborníků se vyjádřil ve stručnosti takto: Problematické je šíření vytvořených map, protože se jedná o přepracovaná data, ostatní výsledky je možné dále šířit.

Následuje kopie otázek, které byly zaslány.

### 6.1. Analýzy 1

První sada analýz spočívala v nalezení spojení mezi základními sídelními jednotkami (dále jen ZSJ). K nalezení nejkratší trasy byly využity zjednodušené úseky. Následně byly tyto nalezené trasy překryty se ZSJ a tak zjištěny ZSJ, kterými vede nalezená trasa. Dotazy na možnosti využití výsledků analýz

1. Mohu předat třetí osobě seznam obsahující ZSJ přes které vede spojení mezi dvěma ZSJ?
2. Mohu k seznamu ZSJ přidat seznam úseků v podobě jejich identifikátorů?

3. Mohu předat mapu vykreslující nalezenou trasu s využitím geometrie úseků, samozřejmě s uvedením zdroje dat?
4. Pomocí dalších překryvných analýz mohu z dat Open Street Map získat geometrii průběhu nalezené trasy. Mohu takto vytvořená data, tedy ta která vzniknou analýzou nad geometrií zjednodušených úseků dále předat?
5. Podobně jako v případě bodu 4 mohu získat geometrii úseků z dat FreeGeoDataCZ (<http://freegis.fsv.cvut.cz/gwiki/FreeGeoDataCZ>), kde je starší verze zjednodušených úseků, kdysi dostupná přímo ze stránek silniční databanky. Mohu takto získaná data (tedy z FreeGeoDataCZ) předat dále?

## 6.2. Analýzy 2

Z WMS služby <http://geoportal.jsdi.cz/> jsem stáhl atributy sčítání dopravy pro jednotlivé úseky a připojil je k zjednodušeným úsekům. Dotazy na možnosti využití výsledků analýz

1. Mohu předat třetí osobě seznam obsahující ZSJ přes které vede spojení mezi dvěma ZSJ a počet vozidel ze sčítání dopravy?
2. Mohu předat mapu intenzity dopravy s využitím Open Street Map nebo úseků z FreeGeoDataCZ?
3. Mohu předat mapu srovnávající nejkratší (nejrychlejší) cesty s intenzitou dopravy s využitím Open Street Map nebo úseků z FreeGeoDataCZ?

Veškerá komunikace proběhla prostřednictvím e-mailu, bez elektronického podpisu. Otázkou tedy zůstává, zda je takové vyjádření vůbec právně závazné a v případě použití dat jej opravdu mohu uplatnit. Dále zůstává otázkou, proč mi tedy data byla poskytnuta pod uvedenou licencí, když s nimi pak mohu v zásadě volně nakládat, kromě jejich přímé distribuce. Další otázkou, která zůstává je zda by daný souhlas dostal kdokoli další. Pokud ano, tak proč není toto přímo uvedeno u dat a data nabídnuta pod takovou licencí ke stažení.

## Závěr

Cílem prezentovaného příspěvku bylo ukázat, že hledání tras v rámci nástroje PostgreSQL/PostGIS je poměrně snadné a může jej realizovat, kdokoli se základní znalostí jazyka SQL. Dále bylo cílem ukázat, že otázka využití dat silniční databanky může být problematické, ikdyž mě osobně bylo povoleno, byť jen prostřednictvím e-mailu uděleno.

**Poděkování.** Tento příspěvek vznikl s pomocí Ředitelství silnic a dálnic ČR a s pomocí nástrojů OpenJUMP, PostgreSQL, PostGIS, pgAdmin, a dalších open source nástrojů.

## Reference

- [1] Landa, M.: *FreeGeodataCZ*, <http://freegis.fsv.cvut.cz/gwiki/FreeGeoDataCZ>, [cit. 8.6.2015].
- [2] Open Street Map community: *Open Street Map*, <http://www.openstreetmap.org>, [cit. 8.6.2015].
- [3] pgRouting Community: <http://pgrouting.org/>, [cit. 8.6.2015].
- [4] Refrations Research: *PostGIS*, <http://postgis.org>, [cit. 8.6.2015].
- [5] Ředitelství silnic a dálnic ČR: *JSDI*, <http://www.jsdi.cz/>, [cit. 8.6.2015].

## Kontaktní adresa

**Ing. Jan Růžička, Ph.D.**, Institut geoinformatiky, VŠB-TUO, 17. listopadu 15, 708 33, Ostrava, Česká Republika,

*E-mailová adresa:* [jan.ruzicka@vsb.cz](mailto:jan.ruzicka@vsb.cz), <http://gis.vsb.cz/ruzicka/>

## BALÍČEK XYPIC: KOMUTATIVNÍ DIAGRAMY

PAVEL STRŽÍŽ (CZ)

**Abstrakt.** Článek představuje několik publikovaných ukázek komutativních diagramů vysázených přes  $\TeX$ ový balíček `xypic`. Ačkoliv lze tento druh diagramů vytvořit v `TikZ`, autor balíčku, Kristoffer H. Rose, na balíčku stále pracuje a je to jistá vhodná alternativa v této konkrétní oblasti sazby.

**Klíčová slova.**  $\TeX$ ,  $\LaTeX$ , `xypic`, komutativní diagramy.

### THE XYPIC PACKAGE: COMMUTATIVE DIAGRAMS

**Abstract.** The article introduces several published commutative diagrams typesetted in the  $\TeX$  package `xypic`. We can typeset these diagrams in `TikZ`, but the author of the package, Kristoffer H. Rose, is actively maintaining the package, therefore it is a fine alternative to the `TikZ` package in this particular field of graphics.

**Key words and phrases.**  $\TeX$ ,  $\LaTeX$ , `xypic`, commutative diagrams.

## 1. Úvod

Zaměřím se na publikované nebo k publikování připravené ukázky, nikoliv na výčet možností balíčku.

S balíčkem jsem se setkal při sazbě časopisů (Czechoslovak Mathematical Journal, Applications of Mathematics a Mathematica Bohemica, více o časopisech viz <https://journals.math.cas.cz/>), kdy se standardně obrázky překreslují v Metapostu či `TikZ`. V tomto případě bylo však lepší prostudovat dokumentaci balíčku, více viz [1].

V práci se výhradně používá jádro  $\TeX$ u, Plain  $\TeX$ . V takovém prostředí se používá načtení přes `\input xypic`, v  $\LaTeX$ ovém prostředí se nejčastěji v preambuli setkáme s příkazem `\usepackage{xypic}`.

## 2. Ukázky

Rovnou představím několik ukázek. Setkal jsem se s nimi při studiu zdrojových kódů dříve publikovaných článků (interní archiv redakce), při úpravě došlých zdrojových kódů a také při překreslování komutativních diagramů, zvláště při užití  $\LaTeX$ ového příkazu `\leftarrow` a `\rightarrow` od Americké matematické společnosti (AMS). Ty se sice do Plain  $\TeX$ u dají dostat, ale poměrně se větví a čitelnost hlavičky článku se znepřehledňuje.

## 2.1. Jednodušší ukázky komutativních diagramů

Na první ukázce je naznačena základní práce s šípkami v diagramu a popisky nad šípkami (`\ar` z angličtiny jako arrow/šipka, `r` jako right/doprava).

$$X_1 \xrightarrow{f_1} X_2 \xrightarrow{f_2} X_3 \xrightarrow{f_3} \dots \xrightarrow{f_{n-1}} X_n \xrightarrow{f_n} \Sigma X_1.$$

```

$$$$\xymatrix{
  X_1 \ar[r]^{f_1} & X_2 \ar[r]^{f_2} & X_3 \ar[r]^{f_3} & \dots \ar[r]^{f_{n-1}} & X_n \ar[r]^{f_n} & \Sigma X_1.
}$$$$

```

Na druhé ukázce vidíme víceřádkový diagram, nastavení šířky sloupců, změnu formátu textu a typu šipek.

$$\begin{array}{ccc}
 N_\psi & \xrightarrow{\text{continuous map}} & M_\varphi \\
 \downarrow & & \downarrow \\
 (N, \mathcal{F}_\psi) & \xrightarrow{\text{induced foliations}} & (M, \mathcal{F}_\varphi)
 \end{array}$$

```

\def\maltxt#1{\textrm{\footnotesize#1}}
$$$$\xymatrixcolsep{20mm}
\xymatrix{
  N_{\psi} & \ar[r]^{\maltxt{continuous}}_{\maltxt{map}} & M_{\varphi} \\
 \downarrow & & \downarrow \\
 (N, \mathcal{F}_{\psi}) & \ar[r]^{\maltxt{induced}}_{\maltxt{foliations}} & (M, \mathcal{F}_{\varphi})
}$$$$

```

Na následující ukázce vidíme změnu typu linek, popisky zleva i zprava u šipek.

$$\begin{array}{ccccccccccc}
 X_1 & \xrightarrow{f_1} & X_2 & \xrightarrow{f_2} & X_3 & \xrightarrow{f_3} & \dots & \xrightarrow{f_{n-2}} & X_{n-1} & \xrightarrow{f_{n-1}} & X_n & \xrightarrow{f_n} & \Sigma X_1 \\
 \varphi_1 \downarrow & & \varphi_2 \downarrow & & \varphi_3 \downarrow & & & & \varphi_{n-1} \downarrow & & \parallel & & \downarrow \Sigma \varphi_1 \\
 Y_1 & \xrightarrow{g_1} & Y_2 & \xrightarrow{g_2} & Y_3 & \xrightarrow{g_3} & \dots & \xrightarrow{g_{n-2}} & Y_{n-1} & \xrightarrow{g_{n-1}} & X_n & \xrightarrow{g_n} & \Sigma Y_1
 \end{array}$$

```

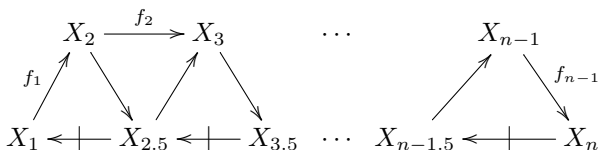
$$$$\xymatrix{
  X_1 \ar[r]^{f_1} \ar[d]_{\varphi_1} & X_2 \ar[r]^{f_2} \ar[d]_{\varphi_2} & X_3 \ar[r]^{f_3} \ar[d]_{\varphi_3} & \dots \ar[r]^{f_{n-2}} & X_{n-1} \ar[r]^{f_{n-1}} \ar[d]_{\varphi_{n-1}} & X_n \ar[r]^{f_n} & \Sigma X_1 \\
 Y_1 \ar[r]^{g_1} & Y_2 \ar[r]^{g_2} & Y_3 \ar[r]^{g_3} & \dots \ar[r]^{g_{n-2}} & Y_{n-1} \ar[r]^{g_{n-1}} & X_n \ar[r]^{g_n} & \Sigma Y_1
}$$$$

```



### 2.2. Komplikovanější ukázky komutativních diagramů

Zde si ukažme komplikovanější skladbu diagramu a linky vedle horizontálních a vertikálních.

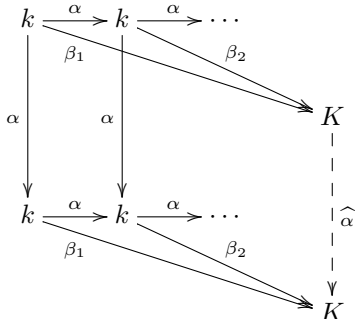


```

 $\xymatrixcolsep{0.2pc}$ 
 $\xymatrix{
& X_2 \ar[dr] \ar[rr]^{f_2} & & X_3 \ar[dr] \\
& & \cdots & & X_{n-1} \ar[dr]^{f_{n-1}} \\
X_1 \ar[ur]^{f_1} & \leftarrow & X_{2.5} \ar[ur] & \leftarrow & X_{3.5} & \cdots & X_{n-1.5} \ar[ur] & \leftarrow & X_n
}$ 

```

Zde ještě jeden vzorek podobného sazečského typu. Např. dd znamená down-down, tedy v diagramu dvakrát dolů, rrd doprava-doprava-dolů atp.

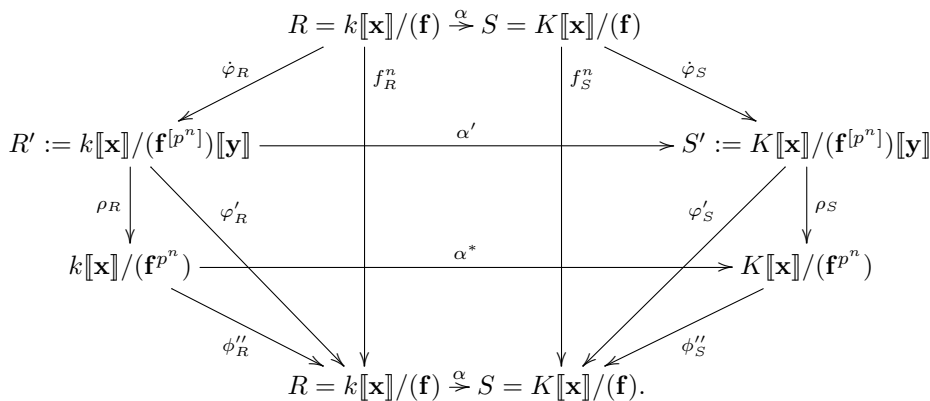


```

 $\xymatrix{
k \ar[r]^{\alpha} \ar[dd]_{\beta_1} & k \ar[r]^{\alpha} \ar[dd]_{\beta_2} & \cdots \\
& & & & K \\
k \ar[r]^{\alpha} \ar[dd]_{\beta_1} & k \ar[r]^{\alpha} \ar[dd]_{\beta_2} & \cdots \\
& & & & K
}$ 

```

Následující ukázka byl pravděpodobně největší komutativní diagram, se kterým jsem se při čištění článků pro tyto časopisy setkal. Principy jsou však stejné jako u jednodušších ukázek, navíc vidíme princip relativního pozicování popisek a zařazení menších TeXových kódů.



```

\def\malL{[![]\def\malR{[]\!]} \def\x{\bold{x}}\def\y{\bold{y}}
\def\matrix @C=2.6mm @R=10mm {
  & R=k\malL \x\malR /(\f)\ar[r]^{\alpha}\ar[ld]_{\dot{\varphi}_R}
  \ar[ddd]^{\lllll\{f^n_R\}}
  S=K\malL \x\malR /(\f)\ar[ddd]^{\lllll\{f^n_S\}}\ar[rd]^{\dot{\varphi}_S}
R':=k\malL \x\malR /(\f^{\{p^n\}})\malL \y\malR \ar[d]_{\rho_R}
  \ar[rrr]^{\alpha'}\ar[rdd]^{\llllllllll\{\varphi_R'\}}\&&&
  S':=K\malL \x\malR /(\f^{\{p^n\}})\malL \y\malR \ar[d]_{\rho_S}
  \ar[lld]_{\llllllllll\{\varphi_S'\}} \
k\malL \x\malR /(\f^{\{p^n\}})\ar[dr]_{\varphi''_R}\ar[rrr]^{\alpha^*}\&&&
  K\malL \x\malR /(\f^{\{p^n\}})\ar[ld]_{\varphi''_S} \
& R=k\malL \x\malR /(\f)\ar[r]^{\alpha}&S=K\malL \x\malR/(\f).
}$$$

```

### 2.3. Komutativní diagramy v textu

V jednom čistěném článku jsem řešil typy šipek. V textu byl jeden typ (balíčky AMS), v diagramech jiné (balíček xypic). Snažil jsem se o sjednocení a zde je ukázka diagramu při zařazení do běžného textu. Upravil jsem si šířku sloupců přes nový příkaz a za použití @1 se upravily parametry doporučené autorem pro tento typ diagramů v textu.

[...] there is an exact sequence  $0 \longrightarrow M \longrightarrow T_0 \longrightarrow \dots \longrightarrow T_m \longrightarrow 0$  with each  $T_i \in \mathcal{C}$  and [...]

```

\def\malwidth{7mm}
[\ldots] there is an exact sequence
  $\xymatrix @C\malwidth @1 {
    0\ar[r] & M\ar[r] & T_0\ar[r] & \cdots\ar[r] & T_m\ar[r] & 0
  }$
with each $T_i \in \mathcal{C}$ and [\ldots]

```

**Poděkování závěrem.** Na tomto místě bych chtěl poděkovat redakci MÚ AV ČR, v.v.i., za souhlas s přetištěním publikovaných a do tisku zařazených obrázků.

### Reference

- [1] ROSE, K. H. – MOORE, R.: xypic – Flexible diagramming macros. Dostupné z <http://ctan.org/pkg/xypic>.

### Kontaktní adresa

**Ing. Pavel Stríž, Ph.D.**, Nakladatelství Martin Stríž, U Škol 940, Bučovice, okres Vyškov, CZ-685 01, Česká republika,  
*E-mailová adresa:* [pavel@striz.cz](mailto:pavel@striz.cz)

## THE MEASUREMENT OF CO<sub>2</sub> BY USING YROBOT PLATFORM

PETER ŠARAFÍN (SK), VERONIKA OLEŠNANÍKOVÁ (SK) AND RÓBERT ŽALMAN (SK)

**Abstract.** Global warming is nowadays a major problem. Global warming is mainly caused by emissions of carbon dioxide. Many organizations are trying to introduce laws and restrictions on the reduction of CO<sub>2</sub> emissions. We decided to monitor CO<sub>2</sub> concentration using open hardware platform Yrobot, which was developed at the University of Žilina.

**Key words and phrases.** CO<sub>2</sub>, measurement, open HW.

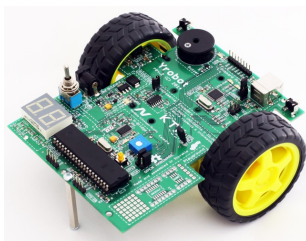
### MERANIE CO<sub>2</sub> S VYUŽITÍM PLATFORMY YROBOT

**Abstrakt.** Globálne otepľovanie je veľkým problémom tejto doby. Globálne otepľovanie je spôsobené emisiami hlavne oxidom uhličitým. Veľa organizácií sa snaží zaviesť zákony a obmedzenia na zníženie CO<sub>2</sub> emisii. Rozhodli sme sa monitorovať CO<sub>2</sub> s využitím otvorenej HW platformy Yrobot, ktorá bola vyvinutá na Žilinskej univerzite.

**Kľúčové slová.** CO<sub>2</sub>, meranie, otvorený HW.

## Introduction

Yrobot (Fig. 1) is a platform used for educational purposes, created on the basis of modular architecture. It was developed at the Department of Technical Cybernetics of Faculty of Management Science and Informatics of University of Žilina [2].



**Figure 1.** Yrobot platform

Designed system represents the Open HW (open hardware) platform which provides the possibility for students to get familiar with fundamentals of electrical

engineering, informatics and computer engineering. Yrobot is supposed to be used as a basis for development of more hardware extensions. One of these extensions is application for CO<sub>2</sub> concentration monitoring in the air.

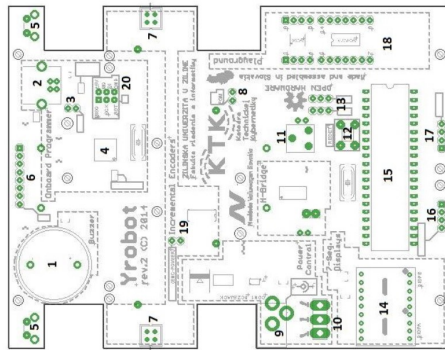
Carbon dioxide is the chemical compound made by two atoms of oxygen and one atom of carbon connected by two double bonds. Carbon dioxide is naturally present in a chemical compounds and in the atmosphere in a certain amount that was produced by animal and plant respiration. After industrialisation, the concentration of carbon dioxide was highly raised because of burning the gas and oils (factories/cars).

Nowadays the concentration in atmosphere is very high. This fact causes health issues for people and even extinction of some animal species because of climatic changes.

Global warming is the significant issue caused by raising concentration of CO<sub>2</sub>, which is one of the greenhouse gasses. It is trapped in the atmosphere and absorbs warmth radiated from the earth surface. So the atmosphere holds the warmth and that is the reason for increasing temperature of the earth, which directly influences the ecosystem.

## 1. Yrobot

In the following figure (Fig. 2), the printed circuit board of the basic Yrobot module can be seen [7].



**Figure 2.** PCB of basic Yrobot module. Key: 1. Buzzer, 2. USB B, 3. SCK Jumper, 4. Self-Programmer MCU, 5. Two buttons, 6. I/O or ADC, V+ and GND, 7. Encoders, 8. Programmer Jumper, 9. Input voltage plug, 10. Power switcher, 11. Trimmer, 12. Reset button, 13. Button Jumper select, 14. 7-Segment display, 15. Main MCU, 16. USART, 17. SPI, 18. Playground area, 19. Encoders jumper power, 20. Self-Programmer MCU SPI

The printed circuit board include four pieces of indicational LED diodes, two 7-segment displays, acoustic buzzer used for signalisation, two motors with integrated gearbox (1:48) and circuitry for sensing the angular turn of the wheels

based on reflex sensors CNY70 [5]. There are also button sensors for barrier detection. The microcontroller (MCU) placed on the board is ATmega16 [3]. It is a low-power CMOS 8-bit MCU based on the AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega16 achieves throughputs approaching 1 MIPS per MHz allowing the system designer to optimize power consumption versus processing speed. The power supply of the system is designed in two ways. User can choose between 7.5 and 10 V adapter and two Li-Pol batteries (3.7 V, 2200 mAh). It is possible to download the program through the integrated USB AVR ISP programmer into the Yrobot's memory.

## 2. Sensor for CO<sub>2</sub> detection

It is possible to choose between two approaches in the CO<sub>2</sub> concentration detection. The first one is measurement using principles about absorption of the wavelength, which is one of the properties of the chemical compounds. This method is called NDIR (nondispersive infrared sensor) [1]. The second method is based on measurement of change of the electrical charge of the chemical reaction. This reaction takes place in the sensor, where particles inside of the sensor are joint with the particles from the air.

We used the sensor with technology of measuring the chemical reaction for our purposes. This method is more financially available as the NDIR.

We selected sensor from the company Figaro with label TGS 4161 [6]. It is a solid electrolyte CO<sub>2</sub> sensor which offers miniaturization and low power consumption. A range of 350–10,000ppm (particles per million) of carbon dioxide can be detected by TGS 4161, making it ideal for indoor air control applications 3.



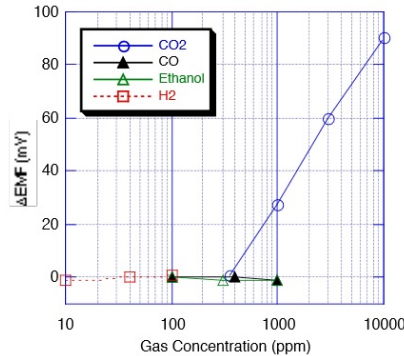
**Figure 3.** TGS 4161 sensor

The CO<sub>2</sub> sensitive element consists of a solid electrolyte formed between two electrodes, together with a printed heater (RuO<sub>2</sub>) substrate. By monitoring the change in electromotive force (EMF) generated between the two electrodes, it is possible to measure CO<sub>2</sub> gas concentration. The top of the sensor cap contains adsorbent (zeolite) for the purpose of reducing the influence of interference gases.

The figure 4 represents typical sensitivity characteristics of TGS 4161. The Y-axis is indicated as  $\Delta\text{EMF}$  which is defined as follows:

$$\Delta EMF = EMF1 - EMF2 \quad (1)$$

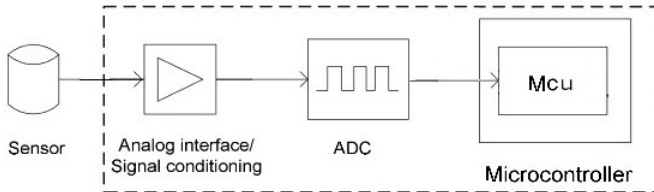
where:  $EMF1 = EMF$  in 350 ppm  $CO_2$ ,  $EMF2 = EMF$  in listed gas concentration.



**Figure 4.** Sensor characteristics

### 3. Interconnecting sensor for $CO_2$ detection with Yrobot

Sensor was interconnected with Yrobot platform using A/D converter (Fig. 5) and measured data were recorded on an SD card. The actual value of  $CO_2$  was also visualized on two 7-segment displays.



**Figure 5.** Block diagram of Yrobot and sensor TGS 4161 attachment

Sensor changed its voltage output corresponding to the concentration of  $CO_2$  in the atmosphere. This signal is connected to the A/D converter's input. The output of the A/D converter is afterwards processed by the microcontroller. Under normal operating conditions, an analogue voltage (concentration of  $CO_2/1000$ ) corresponding to  $CO_2$  concentration is output in the range from 0.4 to 4.0V DC.

## 4. Measurements

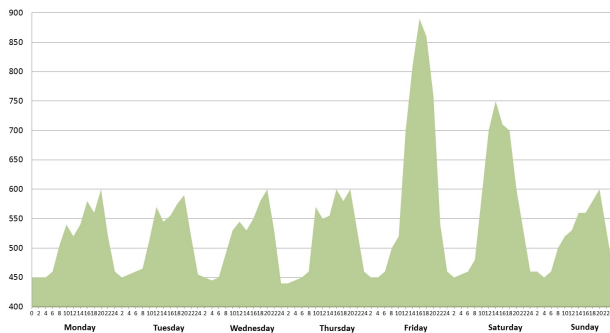
CO<sub>2</sub> gas concentration measurement makes sense in enclosed spaces where the gas can accumulate. As an example of suitable application, the public underground parking lot, where the gas accumulates due to the high density of vehicles in one place can be mentioned. This concentration can also be measured in open areas, mainly near the roads (motorways).

With respect to the platform capabilities, especially since it is not resistant to the weather conditions, we decided to carry out the measurements in enclosed spaces of underground parking lot.

### 4.1. Measurement results

CO<sub>2</sub> is well mixed in the atmosphere, so observations of concentrations from a single site are an adequate indicator of world trends for atmospheric CO<sub>2</sub> [4].

The measured values are in the range from 440 to 900 ppm of CO<sub>2</sub> (Fig. 6). The minimum attainable concentration corresponding to the pure outdoor air is about 420 ppm. For better clarity, the concentration of CO<sub>2</sub> is in this case given in relative terms, where 0 % corresponds to the outdoor air.



**Figure 6.** The data measured in the parking lot

From the measured values it can be seen that the concentration of CO<sub>2</sub> is increased with the arrival of a large number of vehicles. The CO<sub>2</sub> concentration is therefore increased especially on Friday and Saturday when the parking lot is mostly occupied.

## 5. Conclusion

Measured data shows that the concentration of CO<sub>2</sub> in an enclosed parking area does not exceed the value of 900 ppm. The reason of this fact is that the underground parking air conditioning system is well designed and it reduces the air pollution.

When the value rises above the 1200 ppm, we can speak about air pollution of first instance. In this particular case, it is possible to speak about the possibility of human health damage.

There are three stages of air pollution: Stage 1–40 % (800 + 400 ppm), stage 2–65 % (1300 + 400ppm) and stage 3–85 % (1700 + 400ppm). Speaking about the maximum permitted concentration of CO<sub>2</sub> in enclosed spaces, it is usually 2000 ppm (100 %).

The system proposed by us records the pollution status of monitored environment and it stores these data. Data can be further analyzed and processed to serve to appropriate organizations. This system can be also used for providing the information whether the air ventilation system needs to be updated. This system can be complemented by acoustic sensor, so we would be able to monitor the acoustic emissions of given environment. Any other sensors with standard communicating interfaces like TWI, SPI or USART can be used too in the future applications.

## References

- [1] GARCIA-ROMEO, D., FUENTES, H., MEDRANO, N., CALVO, B., MARTINEZ, P. A., AZCONA, C.: *A NDIR-based CO2 monitor system for wireless sensor networks*, Circuits and Systems (LASCAS), 2012 IEEE Third Latin American Symposium, pp.1–4, Feb. 29 2012 – March 2 2012.
- [2] KOCHLÁN, M., HODOŇ, M.: *Open hardware modular educational robotic platform – Yrobot*, Robotics in Alpe-Adria-Danube Region (RAAD), 2014 23rd International Conference, pp. 1–6, 3–5 Sept. 2014.
- [3] ATmega 16 datasheet, <http://www.atmel.com/images/doc2466.pdf>.
- [4] CO2 level monitoring, <http://co2now.org/>.
- [5] CNY 70 datasheet, <http://www.vishay.com/docs/83751/cny70.pdf>.
- [6] Figaro TGS4161 datasheet, [http://www.sos.sk/a\\_info/resource/c/figaro/tgs4161.pdf](http://www.sos.sk/a_info/resource/c/figaro/tgs4161.pdf).
- [7] Yrobot support, <http://www.fribo.sk/yrobot>.

## Contact addresses

**Ing. Peter Šarafín**, Katedra technickej kybernetiky, Fakulta riadenia a informatiky, Žilinská univerzita, Univerzitná 8215/1, 010 26 Žilina, Slovenská Republika,  
*E-mail address:* [Peter.Sarafin@fri.uniza.sk](mailto:Peter.Sarafin@fri.uniza.sk)

**Ing. Veronika Olešnaníková**, Katedra technickej kybernetiky, Fakulta riadenia a informatiky, Žilinská univerzita, Univerzitná 8215/1, 010 26 Žilina, Slovenská Republika,  
*E-mail address:* [Veronika.Olesnanikova@fri.uniza.sk](mailto:Veronika.Olesnanikova@fri.uniza.sk)

**Ing. Róbert Žalman**, Katedra technickej kybernetiky, Fakulta riadenia a informatiky, Žilinská univerzita, Univerzitná 8215/1, 010 26 Žilina, Slovenská Republika,  
*E-mail address:* [Robert.Zalman@fri.uniza.sk](mailto:Robert.Zalman@fri.uniza.sk)



## OPTIMIZATION OF VARIABLES ORDERING IN BINARY DECISION DIAGRAMS USING IMPORTANCE MEASURES

MARTIN VARGA (SK) AND MIROSLAV KVAŠŠAY (SK)

**Abstract.** A binary decision diagram (BDD) is a graph structure proposed for efficient representation of Boolean functions. It is very suitable for representation of and manipulation with a Boolean function containing many variables. However, the main issue is construction of optimal BDDs. The optimality of a BDD mainly depends on the variables arrangement. Finding optimal variables ordering is NP-hard and, therefore, heuristic approaches have to be used to solve this problem. In this paper, we present several new methods for finding a good variables arrangement. The methods are based on some coincidences between Boolean functions and approaches of reliability analysis. An open source software was created to compare these methods.

**Key words and phrases.** Boolean function, binary decision diagram, structural Birnbaum's importance, structural Fussell-Vesely's importance.

## HLADANIE VHODNÉHO USPORIADANIA PREMENNÝCH PRE KONŠTRUKCIU BINÁRNYCH ROZHODOVACÍCH DIAGRAMOV NA ZÁKLADE INDEXOV DÔLEŽITOSTI

**Abstrakt.** Binárne rozhodovacie diagramy (BDD) predstavujú grafovú štruktúru, ktorá slúži pre efektívnu reprezentáciu booleovských funkcií. BDD sú vhodné najmä pre manipuláciu s booleovskými funkciami, ktoré obsahujú veľké množstvo premenných, avšak hlavným problémom je konštrukcia optimálnych BDD, t.j. BDD, ktoré majú minimálnu veľkosť. Optimálnosť BDD závisí predovšetkým od usporiadania premenných. Nájdenie vhodného usporiadania premenných pre konštrukciu optimálneho BDD je NP-ťažká úloha, v dôsledku čoho sa pre riešenie tohto problému využívajú rôzne heuristické prístupy. V tomto článku je predstavených niekoľko nových metód, ktoré môžu byť použité pre nájdenie vhodného usporiadania premenných. Tieto metódy sú založené na koincidencii medzi booleovskými funkciami a niektorými postupmi teórie spoľahlivosti. Pre vzájomné porovnanie jednotlivých metód sme vytvorili open source softvér.

**Kľúčové slová.** Booleovská funkcia, binárny rozhodovací diagram, štruktúrny Birnbaumov index, štruktúrny index Fussell-Vesely.

### Introduction

A Boolean function is a function of several Boolean variables. A Boolean function and a Boolean variable have only two possible values – true and false. If these two values are represented as numbers 1 (true value) and 0 (false value), then the

Boolean function  $f(\mathbf{x})$  has the following form [1]:

$$f(\mathbf{x}) = f(x_1, x_2, \dots, x_n) : \{0, 1\}^n \rightarrow \{0, 1\}, \quad (1)$$

where  $n$  is a number of Boolean variables, i.e. the dimension of the Boolean function,  $x_i$  for  $i = 1, 2, \dots, n$  is a Boolean variable (it has two possible values 0 and 1),  $\mathbf{x} = (x_1, x_2, \dots, x_n)$  is a vector of values of the Boolean variables (a Boolean vector),  $\{0, 1\}^n$  is a space of all Boolean vectors of  $n$  variables and  $2^n$  is the size of the Boolean function, i.e. number of all Boolean vectors of size  $n$ . Please note that a Boolean variable or its negation, i.e.  $x_i$  or  $\bar{x}_i$ , is known as a literal, and a conjunction that contains exactly one literal of every variable of Boolean function  $f(\mathbf{x})$  is called a minterm. Specially, a 1-minterm is a minterm that implies that the function takes value 1, and a 0-minterm is a minterm that implies that the function has value 0.

A Boolean function is monotone or non-decreasing if it meets the following condition [1]:

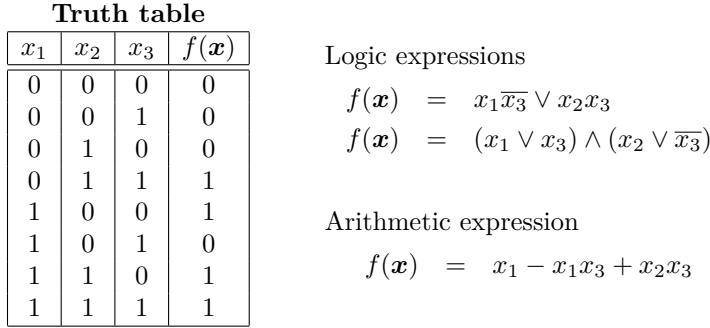
$$\forall \mathbf{x}, \mathbf{y} \in \{0, 1\}^n : \mathbf{x} < \mathbf{y} \Rightarrow f(\mathbf{x}) \leq f(\mathbf{y}), \quad (2)$$

where relation  $\mathbf{x} < \mathbf{y}$  between two Boolean vectors of size  $n$  means that  $x_i \leq y_i$  for all  $i \in \{1, 2, \dots, n\}$  and there exists at least one  $i$  such that  $x_i < y_i$ .

Boolean functions are very important in many engineering fields [1]. They play essential role in the design of logic circuits, in propositional logic and artificial intelligence, in cooperative game theory, in reliability analysis, etc. For such use of Boolean functions, it is necessary to have them in an appropriate form. There exist two basic ways of how to represent Boolean functions (Figure 1). The first of them is truth table. A truth table enumerates all possible combinations of values of input variables and defines the function value for each of them. A truth table can be processed on the computer very simply, but its main disadvantage is its high memory consumption for functions containing many variables. Another possibility is to use mathematical expressions and define the function in the form of logic or arithmetic expression. The main issues of this approach are the fact that a computer realization of mathematical expressions is more difficult and, secondly, it can be more time consuming to compute the function value for a given values of Boolean variables than in the case of truth tables. This implies that these two approaches are not very suitable for computer representation of and manipulation with Boolean functions of high dimensions. Therefore, author of paper [2] has proposed another method for representation of Boolean functions, which is known as a Binary Decision Diagram (BDD).

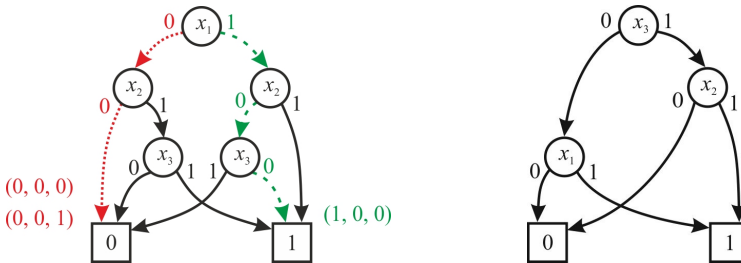
## 1. Binary Decision Diagram

According to [2,3], a BDD is a graph structure that contains two types of nodes: sink and non-sink nodes. Only two sink nodes exist in a BDD. The sink nodes (or leaves) are labeled by numbers 0 and 1, and they coincide with the values of the



**Figure 1.** Basic representations of Boolean functions.

Boolean function expressed as the BDD. The non-sink nodes represent individual variables of the Boolean function. Every non-sink node has two outgoing edges that are labeled by numbers 0 and 1. The edges coincide with two possible values of the corresponding variable. The BDD has also one source node that is denoted as the root. Every path from the root to the 0-labeled (or 1-labeled) sink node represents one or more Boolean vectors of the Boolean function for which the considered function has value 0 (or 1). For example, consider Boolean function  $f(\mathbf{x}) = x_1\bar{x}_3 \vee x_2x_3$  (Figure 1). Some of possible BDDs of this function are shown in Figure 2. The red dotted path in the left diagram begins in the root corresponding to variable  $x_1$ , and it leaves this node through edge labeled by number 0. Then it continues to a node corresponding to variable  $x_2$  and leaves it through edge labeled by 0. Finally, it ends in the 0-labeled sink node. This path corresponds to all Boolean vectors that have value 0 at positions of variables  $x_1$  and  $x_2$  and, therefore, the considered Boolean function has value 0 for Boolean vectors  $(0, 0, 0)$  and  $(0, 0, 1)$ . Similarly, the green dashed path in this diagram agrees with Boolean vector  $(1, 0, 0)$ . Since this path ends in the 1-labeled sink node, the Boolean function has value 1 for the vector  $(1, 0, 0)$ .



**Figure 2.** Examples of BDDs for the Boolean function from Figure 1.

The right picture in Figure 2 shows another BDD that corresponds to the Boolean function from Figure 1. As we can see, this diagram is simpler than the

left one, i.e. it contains fewer nodes and also root-leaf paths are shorter (this implies that we need to make fewer decisions to identify whether the function takes value 1 or 0 for a given Boolean vector). This variety is caused by different arrangement of Boolean variables in individual diagrams, i.e. the variables in the left diagram are arranged from the root to sink nodes in order  $x_1, x_2, x_3$  while, in the right diagram, they are ordered as  $x_3, x_2, x_1$ . The problem of finding variable ordering that results in the creation of a BDD with the fewest number of nodes is NP-hard [4]. One of the possible heuristics that have been proposed to solve this problem for monotone Boolean functions is based on some correlations between approaches related to Boolean functions and methods of reliability engineering [5].

## 2. Reliability Analysis

Reliability is an important characteristic of many systems. Evaluation of system reliability requires creating mathematical model of the system. The model has to define dependencies between system activity and activities of its components (system elements that are assumed to be indivisible into smaller parts). In reliability analysis, we assume that the system activity and activities of its components can be modeled by two different states - functioning (represented by number 1) and failed (presented as number 0). The dependency of system state on the components states is defined by the structure function [6, 7]:

$$\phi(\mathbf{x}) = \phi(x_1, x_2, \dots, x_n) : \{0, 1\}^n \rightarrow \{0, 1\}, \quad (3)$$

where  $n$  is a number of system components,  $x_i$  for  $i = 1, 2, \dots, n$  is a variable defining state of the  $i$ -th system component, and  $\mathbf{x} = (x_1, x_2, \dots, x_n)$  is a vector of components states (state vector). The structure function defines only system topology. If we want to investigate not only topological properties of the system but also some other characteristics, such as system availability, mean time to failure, mean time to repair, etc., the probabilities of individual states of individual system components have to be known [7, 8].

One can see that the formal definition of the structure function corresponds to the definition of a Boolean function. This implies that some techniques related to Boolean functions can be used in reliability analysis [7, 8] and, vice versa, some methods of reliability engineering can be used in the analysis of Boolean functions. However, the second idea has to be realized more carefully because one of the typical assumptions of reliability analysis is a fact that the system structure function is monotone, i.e. there exists no situation in which a failure of any system component can result in system repair. This indicates that most methods of reliability engineering can be applied directly only to monotone Boolean functions.

## 2.1. Importance Measures

One of the main tasks of reliability analysis is identification of components that have the greatest influence on system activity. A lot of measures have been proposed for this purpose [6], e.g. Structural Birnbaum's Importance (SBI), Birnbaum's Importance (BI), Criticality Importance (CI), Structural Fussell-Vesely's Importance (SFVI), Fussell-Vesely's Importance (FVI). Depending on which characteristics of the system are taken into account, these measures can be split into two groups [6]:

- structural importance measures, e.g. the SBI and SFVI – these measures focuses only on system topology;
- reliability/availability importance measures, e.g. the BI, CI, and FVI – they take into account not only system topology but also the probabilities of system components states.

For the purpose of this work, the most important are the SBI and SFVI. The SBI of component  $i$  is defined as a relative number of situations in which a failure (repair) of the considered component results in system failure (repair), i.e. in which the component is critical for system failure (functioning) [6, 7]:

$$\text{SBI}_i = \frac{\sum_{(.i, \mathbf{x}) \in \{0,1\}^{n-1}} (\phi(1_i, \mathbf{x}) - \phi(0_i, \mathbf{x}))}{2^{n-1}}, \quad (4)$$

where  $(.i, \mathbf{x}) = (x_1, x_2, \dots, x_{i-1}, x_{i+1}, \dots, x_n)$  denotes a state vector without the  $i$ -th element, and  $(a_i, \mathbf{x}) = (x_1, x_2, \dots, x_{i-1}, a, x_{i+1}, \dots, x_n)$  for  $a \in \{0, 1\}$ . Clearly, a system component with greater value of the SBI makes difference between system functioning and system failure in more cases than a component with lower value, therefore, it is more important from topological point of view.

The SBI takes into account only situations in which the component is critical for system failure (functioning). However, the component can contribute to system failure (functioning) without being critical [6]. This fact is reflected in the SFVI, which is defined for component  $i$  as the relative number of situations in which the component contributes to system failure or in which the component contributes to system functioning [6]. Before we introduce formulae for computation of this measure, two terms of reliability analysis have to be defined. The first of them is a minimal cut set, which represents a minimal set of components whose simultaneous failure causes that the system fails. Another one is a minimal path set, which agrees with a minimal set of system components whose working ensures that the system will be functioning. Based on this terminology, a component contributes to system failure if at least one minimal cut set containing this component is failed, and it contributes to system functioning if at least one minimal path set containing the component is working [6]. Please note that a minimal cut set is failed if all its component are failed and, similarly, a minimal

path set is functioning if all components presented in it are working. Using minimal cut sets, the SFVI analyzing contribution of component  $i$  to system failure is computed as follows [6]:

$$\text{SFVI}_i^c = \frac{|\{\mathbf{x} : \phi(\mathbf{x}) = 0 \text{ and } \exists \text{MCS}(i) \in \text{MCSs}; \text{MCS}(i) \subseteq \text{CS}(\mathbf{x})\}|}{|\{\mathbf{x} : \phi(\mathbf{x}) = 0\}|}, \quad (5)$$

where MCSs denotes a set of all system minimal cut sets,  $\text{MCS}(i)$  is a minimal cut set containing component  $i$ ,  $\text{CS}(\mathbf{x})$  is a set of all components that are in state 0 in state vector  $\mathbf{x}$ ,  $\{\mathbf{x} : \phi(\mathbf{x}) = 0 \text{ and } \exists \text{MCS}(i) \in \text{MCSs}; \text{MCS}(i) \subseteq \text{CS}(\mathbf{x})\}$  is a set of all state vectors at which component  $i$  contributes to system failure,  $\{\mathbf{x} : \phi(\mathbf{x}) = 0\}$  is a set of all state vectors for which the system is failed, and  $|\cdot|$  denotes size (number of elements) of the argument interpreted as a set.

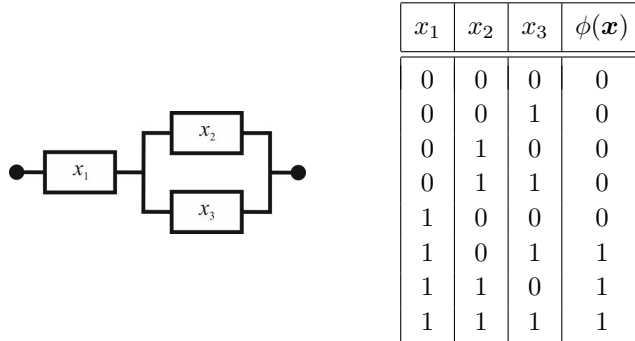
Similarly, the SFVI studying contribution of the component to system functioning is computed based on minimal path sets in the following way [6]:

$$\text{SFVI}_i^p = \frac{|\{\mathbf{x} : \phi(\mathbf{x}) = 1 \text{ and } \exists \text{MPS}(i) \in \text{MPSs}; \text{MPS}(i) \subseteq \text{PS}(\mathbf{x})\}|}{|\{\mathbf{x} : \phi(\mathbf{x}) = 1\}|}, \quad (6)$$

where MPSs denotes a set of all system minimal path sets,  $\text{MPS}(i)$  is a minimal path set containing component  $i$ ,  $\text{PS}(\mathbf{x})$  is a set of all components that are in state 1 in state vector  $\mathbf{x}$ ,  $\{\mathbf{x} : \phi(\mathbf{x}) = 1 \text{ and } \exists \text{MPS}(i) \in \text{MPSs}; \text{MPS}(i) \subseteq \text{PS}(\mathbf{x})\}$  is a set of all state vectors at which component  $i$  contributes to system functioning,  $\{\mathbf{x} : \phi(\mathbf{x}) = 1\}$  is a set of all state vectors for which the system is functioning.

For illustration of the computation of the SBI and SFVI measures consider a simple series-parallel system (Figure 3) whose structure function  $\phi(\mathbf{x})$  corresponds to Boolean expression  $x_1x_2 \vee x_1x_3$ . Based on the truth table in Figure 3 we can see that a failure (repair) of the 1-st component causes system failure (repair) in 3 of 4 possible situations, i.e. component 1 is critical for system failure (functioning) at state vectors  $(., 0, 1)$ ,  $(., 1, 0)$ , and  $(., 1, 1)$ , therefore, its SBI is 0.75. In the similar way, the SBI measures of the remaining components can be computed (Table 1). If we want to compute the SFVI measures, then the minimal cut and path sets have to be known. They can be deduced simply from the diagram in Figure 3. This diagram implies that the system is failed if at least component 1 or at least components 2 and 3 are failed. Therefore, the system has two minimal cut sets, i.e.  $\{1\}$  and  $\{2, 3\}$ . Similarly, the system is functioning if at least components 1 and 2 or at least components 1 and 3 are working, therefore, the minimal path sets are  $\{1, 2\}$  and  $\{1, 3\}$ . This implies that component 1 contributes to system failure in 4 of 5 possible situations (the minimal cut sets containing component 1 cover 4 state vectors of 5 possible in which the system is failed), and it contributes to system functioning in 3 of 3 possible situations (the minimal path sets containing component 1 cover 3 state vectors of 3 possible in which the system is functioning). Therefore, the  $\text{SFVI}_1^c$  is equal to 0.80, and the

SFVI<sub>1</sub><sup>p</sup> equals 1.00. In the similar manner, the SFVI measures for the next two components can be calculated (Table 1).



**Figure 3.** A simple series-parallel system and its structure function.

| Component | SBI <sub>i</sub> | SFVI <sub>i</sub> <sup>c</sup> | SFVI <sub>i</sub> <sup>p</sup> |
|-----------|------------------|--------------------------------|--------------------------------|
| 1         | 0.75             | 0.80                           | 1.00                           |
| 2         | 0.25             | 0.40                           | 0.66                           |
| 3         | 0.25             | 0.40                           | 0.66                           |

**Table 1.** The structural importance measures of individual components of the series-parallel system.

At the end of this section, please note that minimal cut and path sets have equivalents in the terminology of Boolean functions. These equivalents are known as prime implicants and prime implicants respectively. An implicate of Boolean function  $f(\mathbf{x})$  corresponds to a conjunction of literals that implies that the function takes value 0. A prime implicate is an implicate from which no literal can be removed without losing its status as implicate. Similarly, an implicant is a conjunction of literals that implies that the considered function has value 1, and it is prime implicant if no literal can be removed from it without losing its status as implicant. For example, the structure function of the series-parallel system agrees with Boolean function  $f(\mathbf{x}) = x_1x_2 \vee x_1x_3$ . Prime implicants of this function are conjunctions  $x_1x_2$  and  $x_1x_3$  because they imply that the function has value 1 and no literal can be removed from them because after removing any literal we get expressions ( $x_1$  or  $x_2$  or  $x_3$ ) that do not imply that the considered function has value 1. (We can also say that, for example, prime implicant  $x_1x_2$  covers vectors  $(1, 1, 0)$  and  $(1, 1, 1)$  of the function.) Similarly, the prime implicates correspond to conjunctions  $\overline{x_1}$  and  $\overline{x_2x_3}$ . Now, one can realize that minimal path sets correlates with prime implicants and minimal cut sets with prime implicates. However, this is true only for systems whose structure function

is monotonic. In case of systems with non-monotone structure function, minimal cut and path sets are not very common and, instead of them, prime implicants and prime implicates are more often used [9].

### 3. Importance Measures and Binary Decision Diagrams

A BDD can be viewed as a specific form of decision trees. One of the basic ideas that are used in the construction of decision trees is that variables that have greater influence on the output should be stored closer to the root. This idea has been used in [5] to propose a heuristic for finding a variables ordering for construction of a good BDD of a given monotonic Boolean function. The authors have proposed to order variables based on the SBI measures. (Since the structure function of a system can be viewed as a Boolean function, this measure can be used to identify influence of individual Boolean variables on the value of the considered Boolean function.) However, the authors does not consider to use the SFVI. Furthermore, they focused mainly on monotonic Boolean functions since the importance measures have primarily been developed for systems with non-decreasing structure function. Therefore, if we want to use this technique for any Boolean function, then the SBI and SFVI measures have to be generalized for non-monotonic Boolean functions. Based on papers [9, 10], this can be done in the following way:

$$\text{SBI}_i = \frac{|\{(\cdot_i, \mathbf{x}) : f(1_i, \mathbf{x}) \neq f(0_i, \mathbf{x})\}|}{2^{n-1}}, \quad (7)$$

$$\text{SFVI}_i^c = \frac{|\{\mathbf{x} : \exists \text{PIe}(i) \in \text{PIes}; \text{PIe}(i) \Rightarrow f(\mathbf{x}) = 0\}|}{|\{\mathbf{x} : f(\mathbf{x}) = 0\}|}, \quad (8)$$

$$\text{SFVI}_i^p = \frac{|\{\mathbf{x} : \exists \text{PIt}(i) \in \text{PIts}; \text{PIt}(i) \Rightarrow f(\mathbf{x}) = 1\}|}{|\{\mathbf{x} : f(\mathbf{x}) = 1\}|}, \quad (9)$$

where  $\{(\cdot_i, \mathbf{x}) : f(1_i, \mathbf{x}) \neq f(0_i, \mathbf{x})\}$  is a set of all Boolean vectors in which a value of variable  $x_i$  makes difference in the value of the function  $f(\mathbf{x})$ , PIe (PIt) denotes a set of all prime implicates (implicants) of the function  $f(\mathbf{x})$ , PIe( $i$ ) (PIt( $i$ )) is a prime implicate (implicant) containing literal  $x_i$  or  $\bar{x}_i$ ,  $\{\mathbf{x} : \exists \text{PIe}(i) \in \text{PIes}; \text{PIe}(i) \Rightarrow f(\mathbf{x}) = 0\}$  is a set of all Boolean vectors for which the function takes value 0 and that are covered by at least one prime implicate containing literal  $x_i$  or  $\bar{x}_i$ , and  $\{\mathbf{x} : \exists \text{PIt}(i) \in \text{PIts}; \text{PIt}(i) \Rightarrow f(\mathbf{x}) = 1\}$  is a set of all Boolean vectors for which the function has value 1 and that are covered by at least one prime implicant containing literal  $x_i$  or  $\bar{x}_i$ .

For comparison of the proposed methods, we created an open source software (<https://github.com/Kveti/Program-for-creating-BDD>). It implements the ordering based on the exact algorithm (it tries all permutations of variable arrangement), the SBI, and the SFVI computed based on prime implicates. Some results of our testing are presented in Table 2. As we can see, both methods gave



such variables ordering that resulted in creation of optimal BDD, i.e. a BDD with minimal number of internal nodes. This indicates that the idea of using the structural importance measures in creation of a good BDD for Boolean functions of any type (not only for monotonic ones) can be correct, but a lot of other experiments will have to be performed to confirm this hypothesis.

| Function   | Number of internal nodes |     |                   |
|--|--------------------------|-----|-------------------|
|  | Optimal                  | SBI | SFVI <sup>p</sup> |
| $\overline{x_1 x_2} \vee x_3 \vee x_4$                 | 4                        | 4   | 4                 |
| $\overline{x_1(x_2 \vee x_3)} \vee x_4$                | 5                        | 5   | 5                 |
| $\overline{x_1(x_2 \vee x_3)} \vee x_4(x_5 \vee x_6)$  | 13                       | 13  | 13                |
| $\overline{x_1(x_2 \vee x_3)} \vee x_4 \overline{x_5}$ | 11                       | 11  | 11                |
| $\overline{x_1 \vee x_2 x_3 \overline{x_4}}$           | 4                        | 4   | 4                 |

**Table 2.** Examples of experiments for comparison of the heuristics for finding a good variable arrangement for construction of optimal BDDs.

## 4. Conclusion

Boolean functions are very important in many fields of applied science, therefore, it is necessary to have methods that can be used for efficient representation and fast manipulation with them. BDDs are one of the most appropriate approach to solve this issue because they have less memory demands than truth tables and can be processed on computers more efficiently than Boolean expressions. However, the main problem lies in creation of optimal BDDs, i.e. BDDs consisting of minimal number of internal nodes. In this paper, we proposed new methods for solving this problem. They are based on the approach considered in [5], but they can be applied to Boolean functions of any type (not only to monotonic ones). For verification of these methods, an open source software has been created. Using the software, several experiments were done. The experiments showed that our methods can give a good variable ordering. However, the performed experiments could not be used to identify which of our methods is better. Therefore, in the future work, we will have to perform a lot of other experiments that will allow us to solve this problem.

## References

- [1] CRAMA, Y. – HAMMER, P. L.: *Boolean Functions: Theory, Algorithms, and Applications*, New York, NY, Cambridge University Press, 2011, ISBN 0-521-84751-3.
- [2] AKERS, S. B.: *Binary decision diagrams*, IEEE Transactions on Computers, June 1978, vol. C-27, no. 6, pp. 509–516.

- [3] BRYANT, R. E.: *Graph-based algorithms for boolean function manipulation*, IEEE Transactions on Computers, August 1986, vol. C-35, no. 8, pp. 677–691.
- [4] BOLLIG, B. – WEGENER, I.: *Improving the variable ordering of OBDDs is NP-complete*, IEEE Transactions on Computers, September 1996, vol. 45, no. 9, pp. 993–1002.
- [5] BARLETT, L. M. – ANDREWS, J. D.: *An ordering heuristic to develop the binary decision diagram based on structural importance*, Reliability Engineering & System Safety, April 2001, vol. 72, no. 1, pp. 31–38..
- [6] KUO, W. – ZHU, X.: *Importance Measures in Reliability, Risk, and Optimization*, Chichester, UK, John Wiley & Sons, Ltd., 2012, ISBN 1-119-99344-5.
- [7] ZAITSEVA, E. N. – LEVASHENKO, V. G.: *Importance analysis by logical differential calculus*, Automation and Remote Control, February 2013, vol. 74, no. 2, pp. 171–182.
- [8] ZAITSEVA, E. – LEVASHENKO, V. – KOSTOLNY, J.: *Importance analysis based on logical differential calculus and binary decision diagram*, Reliability Engineering & System Safety, June 2015, vol. 138, no. 3, pp. 135–144.
- [9] BEESON, S. – ANDREWS, J. D.: *Importance measures for non-coherent-system analysis*, IEEE Transactions on Reliability, September 2003, vol. 52, no. 3, pp. 301–310.
- [10] KOSTOLNY, J. – KVASSAY, M. – KOVALIK, S.: *Reliability analysis of noncoherent systems by logical differential calculus and binary decision diagrams*, Komunikacie, February 2014, vol. 16, no. 1, pp. 114–120.

## Contact addresses

**Bc. Martin Varga**, Faculty of Management Science and Informatics, University of Žilina, Univerzitná 8215/1, 010 26 Žilina, Slovak Republic,  
*E-mail address:* martivarga@gmail.com

**Ing. Miroslav Kvaššay**, Department of Informatics, Faculty of Management Science and Informatics, University of Žilina, Univerzitná 8215/1, 010 26 Žilina, Slovak Republic,  
*E-mail address:* Miroslav.Kvassay@fri.uniza.sk

## KONVERZE $\text{\LaTeX}$ DOKUMENTŮ DO FORMÁTŮ ELEKTRONICKÝCH KNIH (VYŽIADANÁ PREDNÁŠKA)

**HOFTICH Michal (CZ)**

`tex4ebook` je systém na konverzi dokumentů vytvořených v  $\text{\LaTeX}$ u do formátů elektronických knih. V přednášce si vysvětlíme rozdíly mezi těmito formáty z pohledu potřeb technických a vědeckých publikací. Podíváme se, jak probíhá konverze z  $\text{\LaTeX}$ u za pomoci nástroje `tex4ht`, na kterém je `tex4ebook` založen. Zaměříme se na problémy, které se mohou při konverzi vyskytnout, jak je řešit a jak systém nakonfigurovat pro podporu vlastních příkazů, balíčků, nebo úpravu vzhledu výsledného dokumentu.

## ARÉNA. ROK POTOM

**BIŇAS Miroslav (SK)**

Aréna je platforma pozostávajúca z množiny webových služieb vytvorených za účelom testovania študentských programátorských zadaní. Tento článok opisuje stav projektu a skúsenosti s jeho nasadením po roku od jeho vzniku. Článok konkrétne predstavuje skúsenosti s nasadením Docker kontajnerov s cieľom izolovať potenciálne nebezpečný študentský kód a zvýšiť tak bezpečnosť platformy odolnosťou voči útoku typu injection; nasadenie distribuovanej fronty úloh Cellery s cieľom zvýšenia dostupnosti služby hodnotenia študentských kódov; a predstavuje tiež ďalšie smerovanie projektu.

## DOBRODRUŽSTVÁ V MINECRAFTĚ S PYTHONOM

**BIŇAS Miroslav (SK)**

Jazyk Python je pokladaný za jeden z jazykov s pomerne strmou učiacou sa krivkou, čím sa stáva ideálnym kandidátom pre tzv. prvý jazyk. Prieskum na amerických univerzitách z r. 2014 túto skutočnosť len potvrdil, pretože ukázal, že tento jazyk sa stal dominantným v úvodných programátorských kurzoch. V porovnaní s jazykom Java však neobsahuje toľko nástrojov a prostredí, ktoré by

mohli študentom pomôcť v jednoduchšom porozumení preberaných konceptov a prístupov. Pre jazyk Python je teda problém nájsť alternatívy k nástrojom akým je napr. BlueJ. Za lastovičky je možné pokladať implementácie metodiky robota Karla ako Guido van Robot a Reeborg.

## **SPRÁVA BIBLIGRAFICKÝCH CITACÍ V SYSTÉMU $\text{BIB}_{\text{T}}\text{E}_\text{X}$**

**HALUZA, Pavel, (CZ)**

Príspevek sa zaoberá možnosťami zefektívnenia práce s bibliografickými citaciami v systéme  $\text{BIB}_{\text{T}}\text{E}_\text{X}$ . Pozornosť je venovaná formátu zápisu citácií, preddefinovaným i vlastným štýlom pro jejich sazbu v textu a možnosťou správy citačnej databázy.

## **HYBRIDNÝ MULTIROBOTICKÝ SYSTÉM**

**CHOVANEK, Michal (SK); ČECHOVIČ Lukáš (SK)**

V príspevku bude prezentovaný hybridný multirobotický systém. Pozostáva z minirobotov (35 mm × 35 mm) a virtuálneho prostredia, v ktorom sa vykonávajú experimenty – veľkoplošný monitor. Je tak umožnená rýchla modifikácia experimentu a overovanie simulácií na skutočných robotoch.

## **SMART TV, ANDROID TV, IPTV**

**JURKULÁK, Milan, (SK)**

V prezentácii predstavím históriu a vývoj TV aplikácií viacerých technológií: SmartTV, Android TV. Stručný prehľad framework-ov: naPanasonic, Samsung, LG, Philips, vývoj aplikácií preSmart TV. Príklady a ukážky skriptovania. Stručný rozbor porúch, častých problémov, použitých protokolov. Predstavené technológie: Leanback support library v7+, Gradle build systém, RecyclerView. Na záver ukážková Android TV aplikácia, popis UI.

## **IPython (JUPYTER) NOTEBOOK, SILNÝ POMOCNÍK PRE VZDELÁVANIE**

**KAUKIČ, Michal (SK))**

Ukážeme, ako sa dá webový prehliadač s IPython notebookmi použiť pri učení. Chceme inšpirovať potenciálnych používateľov na samostatné experimentovanie. Bude to nielen vysvetľovanie, ale každý bude mať možnosť si notebooky vyskúšať.

## VÝVOJ REDAKČNÉHO SYSTÉMU PRE KONFERENCIU

**KOMPIŠ, Jakub; STROMČEK, Adam; ŠÁLEK Tomáš, (SK)**

Cieľom projektu je vytvorenie online informačného systému pre konferencie pomocou využitia dostupných Open Source technológií. Projekt sa primárne zaoberá potrebami medzinárodnej konferencie OSSConf, ale jedným z cieľov je implementácia univerzálneho Open Source informačného systému použiteľného aj pre potreby iných konferencií. Informačný systém má obsahovať dva hlavné moduly, a to recenzný modul a modul slúžiaci na organizovanie podporných služieb konferencie, ako sú napríklad strava či ubytovanie účastníkov.

## 3D TLAČIAREŇ

**KUZMIAK, Dávid, (SK)**

Posledné roky zažívajú 3D tlačiarne boom na trhu. Majú širokospektrálne využitie v rôznych oblastiach. Od kovových výtlačkov až po plastové výtlačky. Väčšinou ide o špeciálne dielce, ktoré sa neoplatí robiť inou technológiou lebo je treba len zopár kusov. V dnešnej dobe prerazili aj 3D tlačiarne dostupné do domácnosti. Robí sa niekoľko lacných koncepcií 3D tlačiarní. Takmer všetky tieto lacné tlačiarne sú open-source (RepRap). RepRap je slobodná (open source hardware), samoreplikujúca sa 3D tlačiareň. Názov RepRap pochádza z anglického replicating rapid prototyper. V príspevku predstavím doma postavanú open-source 3D tlačiareň koncepcie Prusa.

## SÉRIOVÁ KONZOLA NA RASPBERRY Pi

**MAJER, Tomáš, (SK)**

V príspevku ukážem postup ako si zapojiť a používať sériovú konzolu na Raspberry Pi. Konzola je veľmi užitočná, ak nemáte Raspberry Pi pripojené k monitoru a klávesnici a nefunguje počítačová sieť...

## DIGITYP

**NOGA, Martin; MANDÚCHOVÁ Katarína, (SK)**

Projekt DigiTyp má za úlohu uľahčiť učiteľom základných a stredných škôl testovanie svojich študentov. Výhodou je, že vďaka tomuto projektu budú môcť učitelia medzi sebou zdieľať testové otázky. Tým bude zvýšená efektivita ich práce. Z pripravených otázok bude možné vytvoriť test priamo na stránke a testovať žiakov online. Taktiež bude možné tieto testy pomocou L<sup>A</sup>T<sub>E</sub>X-u skompilovať do

formátu pdf a vytlačiť pre offline testovanie. Dôraz je kladený najmä na používateľsky príjemné rozhranie, ktoré zvládnu aj nie veľmi technicky zdatní učitelia.

V projekte sa používajú technológie, ktoré sú stále vyvíjané a aktuálne. Aplikáciu vyvíjame pomocou JavaServer Faces JSF, pre ktorú sú dostupné rozsiahle knižnice komponentov. V našom projekte používame knižnicu PrimeFaces. Na prístup k databáze používame framework Hibernate. Ako databázu sme zvolili MySQL. Vďaka tomu, že všetky z uvedených technológií sú voľne šíriteľné, celý projekt môžeme vyvíjať ako open source, čo umožní jeho ľahké šírenie a prispôbovanie.

## OTVORENÝ MAPOVÝ PORTÁL WEBMAP.SK PRE ŠTUDENTSKÉ APLIKÁCIE A GEODÁTA

**OFÚKANÝ, Miloslav; ABRAHÁM, Dominik; POLÁKOVÁ, Radka;  
KALIVODA, Martin; PANÁK Martin, (SK)**

Portál Webmap.sk beží na virtuálnom serveri (VPS) s operačným systémom Debian 7. Každý používateľ má v ňom vytvorené konto, vyhradený priestor a prístup cez SSH a FTP. Na strane servera je podpora programovacích jazykov, ako napríklad PHP, Python (verzie 2 aj 3) či Java (OpenJDK). Webový server Apache je konfigurovaný tak, že každý používateľ môže svoje súbory sám publikovať na webe a sú dostupné na adrese <http://webmap.sk/~<username>>. Databázový systém PostgreSQL disponuje rozšíreniami PostGIS a pgRouting. Pristupovať k nemu je možné aj vzdialene cez pgAdmin. Používateľ môže tiež využiť GeoServer spolu s niekoľkými vybranými rozšíreniami, vďaka čomu môže využívať populárne mapové služby (WMS, WFS, WCS), katalógovú službu (CSW) a procesnú službu (WPS). Ďalší softvér poskytujúcou zdieľaný geoprocessing je 52°North, ktorý okrem vlastnej knižnice sprostredkúva algoritmy softvérov Sextante a GRASS 7. K dispozícii je aj ďalší voľne šíriteľný softvér, vhodný pre vývoj geografických informačných systémov.

V súčasnosti poskytuje Webmap hosting pre niekoľko študentských aplikácií. Úlohou aplikácie Cyklorouting je pomôcť cyklistom pri plánovaní cyklotrás. Užívateľia majú možnosť zvoliť si typ trasy - najkratšiu, optimalizovanú pre horského cyklistu, optimalizovanú pre cestného cyklistu. Po vložení počiatočného a koncového bodu aplikácia zrealizuje výpočet trasy. Výsledná cesta sa zobrazí v mape. Túto trasu je možné modifikovať pridaním prechodových bodov a stiahnuť si ju do formátu GPX, čo umožňuje vložiť cyklotrasu do navigácie. Realizáciou pgRouting workshopu vznikla webová mapová aplikácia s funkcionalitou vyhľadania najkratšej vzdialenosti medzi dvojicou bodov. Operuje nad vrstvou dopravných komunikácií v SR, pričom zohľadňuje typ komunikácie. Aplikácia Hiking poskytuje prehľad turistických trás, ktoré sú dynamicky načítavané podľa aktuálneho rozsahu mapového okna. Výberom v zozname je možné sa priblížiť na vybranú

trasu. Výberom trasy používateľ získa detaily o nej. Zdieľaný geoprocessing demonštruje služba WPS, ktorá umožňuje využívať algoritmy GRASS 7, Sextante či GeoServeru v rozhraní webovej mapovej aplikácie.

## **LABORATOŘ INFORMAČNÍCH SYSTÉMŮ A PROCESNÍHO MODELOVÁNÍ**

**RÁBOVÁ, Ivana; PRIŠČÁKOVÁ, ZUZANA; KLEGOVÁ, Jaroslava,  
(CZ)**

Cílem příspěvku je seznámit s inovací infromatické výuky na PEF Mendelovy univerzity v Brně. V rámci příspěvku bude pojednáno o struktuře výuky a podpůrném softwaru, který je implementován zdarma, nebo za univerzitní licence. Konkrétní ekonomická stránka nebude uvedena, dodavatelské firmy si nepřejí zveřejňovat jednotlivé finanční položky. Laboratoř byla vybudována ze zdrojů MŠMT. V rámci investičního projektu Vybudování laboratoře informačních systémů a procesního modelování na PEF Mendelovy univerzity byla v roce 2014 realizována speciální učebna, ve které je nainstalováno velké množství zajímavého softwaru. V dalších dvou letech pak byly a jsou implementace průběžně rozšiřovány a v rámci navazujících projektů pak tvořeny výukové materiály a podpory k jednotlivým předmětům. Software je zaměřen na tři oblasti. Jednu tvoří skupina CASE nástrojů, druhou část softwarového vybavení nové učebny pak tvoří informační systémy typu ERP a konečně třetí oblast tvoří projektové řízení.

## **MOŽNOSTI 3D VIZUALIZACÍ V GRASS GIS**

**RŮŽIČKOVÁ, Kateřina, (CZ)**

GRASS GIS je využíván především pro širokou nabídku analytických funkcí. Vizualizace výstupů je většinou považována za druhotnou a méně podstatnou stránku funkcí GRASS. Nicméně i zde může tento program konkurovat komerčním programům. Pro 3D vizualizace má k dispozici nástroje NVIZ, které jsou kontinuálně zdokonalovány. Tento příspěvek bude zaměřen na možnosti práce s NVIZ v oblasti 3D statických i dynamických vizualizací geodat a tvorbu výstupů z těchto vizualizací.

## **PROGRAMOVANIE PRE STREDNÉ ŠKOLY**

**ŠECHNÝ, Martin, (SK)**

Programovanie – zručnosť pre 21. storočie – jedna z aktuálnych priorit vzdelávania v EÚ. Prečo potrebujeme viac programátorov? Prečo a ako modernizovať

súčasné vyučovanie programovania na stredných školách? Na základe skúseností zo škôl, firemných kurzov a praxe som prepracoval obsah a metodiku vyučovania algoritmizácie a programovania pre stredné technické školy v programovacích jazykoch C/C++ a pripravujem menej technicky náročnú alternatívu s programovacím jazykom Python. Predstavujem učebný text s príkladmi vo formátoch pdf a html. Iné alternatívy programovania? Webdizajn a php, GNU/Linux a bash. . .

## **ROZŠÍRENIE WEBOVEJ APLIKÁCIE T<sub>E</sub>X onWeb**

**ŠINO, Peter; VRTIAK, Peter, (SK)**

Aplikácia T<sub>E</sub>XonWeb ponúka možnosti vytvárania dokumentov v typografickom systéme TeX aj pre menej skúsených používateľov. Naším cieľom bolo vytvoriť dva doplnky – nástroj pre tvorbu prezentačných materiálov v triede Beamer a kresliaci editor pre kreslenie základných geometrických útvarov.

Nástroj pre tvorbu prezentácie umožňuje vyplňať základné atribúty prezentácie do textových polí. Tieto informácie aplikácia spracuje a vygeneruje zdrojový kód. Používateľ nemusí poznať príkazy v triede Beamer ale dokáže si takýmto spôsobom vytvoriť vlastnú prezentáciu.

V nástroji pre kreslenie sme doposiaľ vytvorili jednoduchý editor pre kreslenie geometrických útvarov (čiara, kruh, štvorec) pomocou myši. Aplikácia vygeneruje zdrojový kód v príkazoch balíčka TikZ. V budúcnosti budú podporované viaceré útvary.

## **NOVINKY V LINKOVANÝCH GEODÁTACH A DOPYT TRHU A PONUKA VZDELÁVANIA V OBLASTI GEOINFORMATIKY V EURÓPE**

**TUCHYŇA; Martin, (SK)**

Prepojené otvorené geopriestorové údaje svojou povahou ponúkajú možnosť vytvárania obsahových, alebo polohových väzieb tak s údajmi priestorovej i nepriestorovej povahy využitím potenciálu sémantického webu. Informácie o prvých príkladoch vybraných implementácií, spolu s výsledkami prieskumu projektu GI-N2K (Geographic information Need To Know) zameraného na lepšie porozumenie dopytu i ponuky vzdelania a školení v oblasti geoinformatiky v Európe prinesie predmetný príspevok.



Vydanie tohto zborníka bolo podporené grantom slovenskej kultúrno-edukačnej agentúry KEGA č. 011ŽU-4/2014 „Experimentálna matematika – zviditeľnenie neviditeľného“.

**OTVORENÝ SOFTVÉR VO VZDELÁVANÍ, VÝSKUME A V IT RIEŠENIACH**  
Zborník príspevkov medzinárodnej konferencie OSSConf 2015,  
konanej 1.–3. júla 2015 v Žiline

Prvé vydanie 2015

Elektronická sadzba programom pdfL<sup>A</sup>T<sub>E</sub>X Rudolf Blaško

Vydala Žilinská univerzita v Žiline

Tlač EDIS – vydavateľstvo Žilinskej univerzity v Žiline

Náklad 100 ks

ISBN 978-80-970457-7-7

ISBN 978-80-970457-7-7



9 788097 045777