

UNIVERZITA KONŠTANTÍNA FILOZOFA V NITRE
FAKULTA PRÍRODNÝCH VIED

**TVORBA NÁSTROJOV V GRASS GIS PRE
PODPORU KRAJINNO-EKOLOGICKÉHO
VÝSKUMU**

DIPLOMOVÁ PRÁCA

2015

Bc. Michal Ševčík

UNIVERZITA KONŠTANTÍNA FILOZOFA V NITRE
FAKULTA PRÍRODNÝCH VIED

**TVORBA NÁSTROJOV V GRASS GIS PRE
PODPORU KRAJINNO-EKOLOGICKÉHO
VÝSKUMU**

DIPLOMOVÁ PRÁCA

Študijný program: Environmentalistika
Študijný odbor: 4.3.1 ochrana a využívanie krajiny
Školiace pracovisko: Katedra ekológie a environmentalistiky
Školiteľ: Mgr. Imrich Jakab, PhD.

2015

Bc. Michal Ševčík



Univerzita Konštantína Filozofa v Nitre
Fakulta prírodných vied

ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta: Bc. Michal Ševčík
Študijný program: environmentalistika (Jednoodborové štúdium, magisterský II. st., denná forma)
Študijný odbor: 4.3.1 ochrana a využívanie krajiny
Typ záverečnej práce: Diplomová práca
Jazyk záverečnej práce: slovenský
Sekundárny jazyk: anglický

Názov: Tvorba nástrojov v GRASS GIS pre podporu krajinno-ekologického výskumu
Anotácia: Práca je zameraná na vytváranie nástrojov pre podporu krajinno-ekologického výskumu využitím Open Source GIS desktop aplikácie GRASS GIS, programovacieho jazyka Bash pod operačným systémom Linux. Výstupom práce budú konkrétne otestované nástroje, ich podrobná dokumentácia a príklady ich použitia v krajinno-ekologickom výskume.

Školiteľ: Mgr. Imrich Jakab, PhD.
Oponent: Mgr. Henrich Grežo, PhD.
Katedra: KEE - Katedra ekológie a environmentalistiky

Dátum zadania: 26.10.2013

Dátum schválenia: 28.10.2013

prof. RNDr. Juraj Hreško, CSc.
schválil/a

Pod'akovanie

Touto cestou by som sa rád poďakoval svojmu školiteľovi Mgr. Imrichovi Jakabovi, PhD., za vedomosti a čas, ktorý mi venoval v posledných dvoch rokoch, a rovnako tak aj mojej priateľke za trpezlivosť, ktorú so mnou mala počas celého štúdia.

ABSTRAKT

ŠEVČÍK, Michal: Tvorba nástrojov v GRASS GIS pre podporu krajinno-ekologického výskumu. [Diplomová práca]. Univerzita Konštantína Filozofa v Nitre. Fakulta prírodných vied. Školiteľ: Mgr. Imrich Jakab, PhD. Stupeň odbornej kvalifikácie: Magister v odbore environmentalistika. Nitra : FPV, 2015. 88 s.

Práca je zameraná na tvorbu nástrojov s využitím v rôznych oblastiach ekológie a environmentalistiky. Výsledkom práce je deväť skriptov rozdelených do štyroch kategórií podľa hlavných možností využitia: Výpočet reálnej rozlohy; DBMS a tvorba priestorových údajov; Výpočet viditeľnosti pomocou paralelných výpočtov; Simulácia zaplavenia územia. Práca obsahuje opis všetkých algoritmov, ich testovanie, vyhodnotenie a porovnanie spolu s charakteristikou vstupov, výstupov a možnosťami využitia. Vzhľadom na open-sourcovú logiku, pod ktorou boli skripty vytvárané, sú súčasťou práce všetky zdrojové kódy aktuálnych verzií skriptov. Všetky skripty sú vytvorené v skriptovacom jazyku BASH s využitím GRASS GIS a jeho analytických a štatistických nástrojov.

Kľúčové slová: GRASS GIS, tvorba nástrojov, BASH, open-source, rozloha, viditeľnosť, databáza, simulovanie záplav

ABSTRACT

ŠEVČÍK, Michal: GRASS GIS tools creation for supporting the lanscape-ecological research. [Diploma Thesis]. Constantine the Philosopher University in Nitra. Faculty of Natural Sciences. Supervisor: Mgr. Imrich Jakab, PhD. Degree of Qualification: Master of environmentalism. Nitra: FNS, 2015. 88 p.

The thesis is focused on development of tools using in different fields of Ecology and Environmental Sciences. The result of this thesis is nine scripts divided into four categories based on main possibility of using: Real area computation; DBMS and creating of spatial data; Viewshed analysis using parallel computation; Simulation of flooding area. The thesis contains a description of all algorithms, their testing, evaluation, comparison with characteristic of inputs, outputs and potential. Considering to open-source logic creation of scripts, all source codes of current script's version are part of the thesis. All scripts are created in BASH scripting language, using GRASS GIS and its analytic and statistic tools.

Keywords: GRASS GIS, tools development, BASH, open-source, area, viewshed analysis, database, flooding simulation

OBSAH

ÚVOD A CIEĽ	7
1 PREHĽAD RIEŠENEJ PROBLEMATIKY	8
1.1 Definícia GIS.....	8
1.2 Priestorové údaje	12
1.3 Dátové modelovanie	15
1.3.1 Vektory.....	16
1.3.2 Rastre.....	17
1.4 Zber údajov.....	19
1.5 Priestorové analýzy.....	20
2 METODIKA PRÁCE	26
2.1 Opis vzniku skriptov.....	26
2.2 Tvorba skriptov	27
2.3 Štruktúra skriptov	28
2.4 Testovanie skriptov	29
2.5 Štruktúra opisu skriptov v práci	30
3 VÝSLEDKY A DISKUSIA	32
3.1 Opis spoločných častí skriptov.....	32
3.2 Výpočet reálnej rozlohy.....	35
3.3 DBMS a tvorba priestorových údajov	44
3.4 Výpočet viditeľnosti pomocou paralelných výpočtov.....	56
3.5 Simulácia zaplavenia územia.....	66
ZÁVER	79
ZOZNAM POUŽITEJ LITERATÚRY	81
PRÍLOHY	88

ÚVOD A CIEĽ

Geografické informačné systémy (GIS) patria k najdynamickejšie sa rozvíjajúcim informačným systémom vôbec. Schopnosť prepojiť údaje na polohovo lokalizované goeobjekty otvára nové možnosti využitia nie len v geografii, ale aj v rôznych iných odvetviach a činnostiach, ako napríklad simulačné modely šírenia vody či ohňa, pri analýzach kriminality alebo zisťovaní zdravotnej dostupnosti.

Neustály rozvoj všetkých odvetví geoinformačných technológií, ako diaľkový prieskum Zeme, družicové polohové systémy, digitálna fotogrametria, lidar, či digitálne modely reliéfu, umožňuje čím ďalej, tým kvalitnejšie a presnejšie získavanie údajov. Zo zvyšujúcou sa presnosťou narastá aj veľkosť a množstvo týchto údajov a manuálne spracovanie je často potrebné nahradiť automatizáciou. Aj s toho dôvodu trh s GIS technológiami a službami každým rokom rastie.

Jedným z mála plne programovateľných, open-sourcových GIS softvérov je aj GRASS GIS. Patrí bezkonkurenčne k najstarším a najmocnejším GIS softvérom vo svojej triede. Schopnosť ovládať GRASS GIS príkazovým riadkom umožňuje automatizáciu pomocou väčšiny programovacích či skriptovacích jazykov (BASH, Python, C, Java atď.). Užívateľ tak môže vytvárať nové neexistujúce nástroje, poprípade upravovať už existujúce nástroje podľa vlastnej potreby a uľahčiť tak prácu sebe i ostatným.

„The application of GIS is limited only by the imagination of those who use it.”

Jack Dangermond

Cieľom diplomovej práce je vytvorenie softvérových nástrojov v GRASS GIS využitelných v rôznych oblastiach krajino-ekologického výskumu.

1 PREHLAD RIEŠENEJ PROBLEMATIKY

1.1 Definícia GIS

V rozvoji modernej spoločnosti zohrávajú informačné technológie v posledných rokoch rozhodujúcu úlohu (Majerník, 2008). Jedným z odvetví informačných technológií sú aj geografické informačné systémy (GIS). I keď majú GIS široký interdisciplinárny charakter (využitie v geografii, matematike, kartografii, informatike, DPZ atď.), sú spájané najmä s geoinformatikou - vednou disciplínou zaoberajúcou sa teoretickým a metodickým aspektom spracovania geografických informácií (Hofierka, 2003).

Pre lepšie pochopenie termínu geografické informačné systémy uvádza Tuček (1998) jeho nasledovné rozloženie:

- **geo** – vyjadruje, že GIS pracujú s údajmi a informáciami, ktoré majú určitý vzťah k Zemi (buď priamo na povrchu v jej blízkosti, alebo pod povrchom) a sú lokalizované v akomkoľvek súradnicovom systéme v priestore, poprípade v čase,
- **grafický** – na zobrazovanie výsledkov, prezentácii dát, či komunikácii s užívateľom využíva grafické prostriedky,
- **informačný** – GIS uskutočňujú zber, ukladanie, analýzu a syntézu rôznych údajov za účelom získavania nových, ktoré sa následne využívajú ako podklady v rôznych činnostiach (plánovanie, modelovanie, riadenie, rozhodovanie atď.),
- **systém** – GIS spájajú do jedného funkčného celku viacero údajov, pracovných postupov, technické a programové vybavenie, personál, ako aj užívateľov.

GIS ako také majú veľmi široké možnosti využitia, a preto je veľmi obtiažne ich presne definovať (Escobar et al., 1999). Existuje mnoho rôznych prístupov a hľadísk ich chápania, ktoré viacerí autori ako napr. Komárková, Kopáčková (2008); Rapant (2002); Tuček (1998); Voženílek (1998) a ďalší, triedia do troch rôznych úrovni ponímania:

a) GIS ako softvér

Na tejto "najnižšej" úrovni chápeme GIS ako určitý programový produkt, o ktorom sa obecne vyhlasuje, či tento produkt je alebo nie je GIS. Takáto interpretácia GIS je nesprávna. Takýmto chápaním sú GIS definované napríklad podľa:

- Patra (2013): „GIS je počítačový systém, ktorý pridáva množstvo kvalít a vlastností zemepisnej polohy a pomáha pri plánovaní a rozhodovaní.“, alebo

- ESRI (2012): „GIS je počítačový softvér, ktorý spája geografické informácie (kde objekty sú) s popisnými informáciami (aké objekty sú), pričom na rozdiel od papierových máp môže predstavovať mnoho vrstiev rôznych informácií.“

b) GIS ako konkrétna aplikácia

GIS sú chápané ako komplexné funkčné informačné systémy slúžiace na prácu s konkrétnymi údajmi určitého územia s určitým cieľom, so sociálnym a organizačným zázemím. Pod touto úrovňou rozumieme hardvér, softvér, údaje, postupy a ľudí, ako na obsluhu, tak aj na používanie. Príkladom môžu byť nasledovné definície:

- Jakubík (2010): „GIS - informačné systémy zamerané na zber, ukladanie, aktualizáciu, vyhodnocovanie a najmä analýzy priestorových informácií tabuľkovou alebo kartografickou formou.“

- Goodchild (2001): „GIS - využívajú počítačovú technológiu na zachytávanie, ukladanie, manipuláciu, analyzovanie, modelovanie a zobrazovanie informácií o povrchu Zeme a javoch na nej.“

c) GIS ako informačné technológie.

Je to najkomplexnejšie chápanie pojmu GIS, a tým pádom aj najťažšie definovateľné. Ide o celkové prostredie, v ktorom GIS aplikácie vznikajú. Príkladom môže byť definícia:

- Ramachandran (2010): „GIS je súbor hardvéru, softvéru a procedúr s cieľom uľahčiť riadenie, manipuláciu, analýzy, modelovanie, reprezentáciu a zobrazovanie georeferenčných údajov na riešenie zložitých problémov s ohľadom na plánovanie a riadenie zdrojov“.

Hlavnou náplňou GIS je spracovanie a práca s geografickými (priestorovými) údajmi. Avšak GIS v sebe skrývajú omnoho viac a ich využitie je obmedzené len údajmi a schopnosťami užívateľa.

Využívajú sa v odvetviach ako (Rapant, 2005; Voženílek, 1998):

- Ochrana životného prostredia: monitorovanie erózií, hluku, kvality ovzdušia atď.,
- Štátna správa a samospráva: územné plánovanie, správa majetku, sčítanie ľudu atď.,
- Poľnohospodárstvo: evidencia pôd, lesné plánovanie, plánovanie meliorácie pôd, umiestňovanie rastlinnej aj živočíšnej výroby atď.,
- Obchod: využite demografických a socioekonomických údajov pre čo najefektívnejší výber miesta pre nový obchod, továreň, reštauráciu atď.,
- Doprava: plánovanie cesty, navigácia, optimalizácia prepravy na dlhšie trasy, atď.,
- Obrana: rozmiestnenie vojsk, organizácia vojenskej prepravy, vzdialená navigácia atď.

- Archeológia: vizualizácia miest nálezov, vyhľadávanie potencionálne nových lokalít atď.

Vďaka širokému užívateľskému zastúpeniu existuje veľké množstvo programových balíkov (niekoľko 100 –viek softvérov) (Longley et al., 2005). Nato aby mohol byť softvér označený ako GIS softvér musí spĺňať nasledujúcu štruktúru podsystémov (Voženilek, 1998; Huisman, Rolf, 2009):

1. Vstup údajov: premena a overenie informácií do systémom kompatibilnej formy.
2. Spracovanie a uloženie údajov: zaisťuje správnosť a prípravu údajov na ďalšie použitie.
3. Priestorové analýzy: hlavná časť softvéru, ktorá GIS odlišuje od iných softvérov. Ide o analýzy a syntézy za použitia priestorových vzťahov, z ktorých vznikajú nové údaje.
4. Transformácia a konverzia údajov: spočíva v zmene formy, bez zmeny obsahu. Úlohy ako reklasifikácia, či zmena mierky atď.
5. Výstupy a prezentácie údajov: v tomto kroku sa výsledky analýz spracovávajú do konečnej podoby, a to formou máp, tabuliek, grafov, diagramov, atď.
6. Užívateľské rozhranie: jeho význam spočíva v interpretovaní príkazov užívateľa do systémom pochopiteľnej formy.

GIS softvéry je možné rozdeliť na základe viacerých hľadísk:

- Podľa funkčnosti a typu ich Longley et al. (2005) rozdelili na:
 - a) Desktopové:** Najrozšírenejší typ, využívaný skôr na prezentáciu údajov a menej na ich zber. Pokrývajú širokú škálu využitia, na základe čoho ich Koreň (1995) rozdeľuje na:
 - kartografické: využitie hlavne na vizualizáciu a kartografickú prezentáciu priestorových údajov,
 - databázové: dôraz na tvorbu, editáciu, prepájanie, dopytovanie, či uchovávanie priestorových údajov,
 - analytické: hlavné využitie analytických prostriedkov (modelovanie v hydrológii, biológii, geológii atď.).

Špeciálnym typom desktopových softvérov sú aj tzv. developerské aplikácie, ktoré sú určené najmä pre vývojárov, poprípade skúsených programátorov a umožňujú prispôsobiť GIS pre jednotlivé potreby užívateľa, či možnosť spojenia s inými softvérovými systémami.

b) Serverové: GIS bežiacie na počítačových serveroch, ktoré sú schopné splniť požiadavky od vzdialene pripojených užívateľov. Zvyčajne sú zamerané na zobrazovanie

a dopyt na databázu, ale v súčasnosti ich možno využívať aj na mapovanie, trasovanie a iné činnosti.

c) Mobilné (hand-held): Zvyčajne jednoduchšie zariadenia, určené na zber priestorových informácií.

d) Ostatné: Ide o rôzne komerčné a nekomerčné softvéry, ktoré dokážu spracovávať niektoré z typov GIS súborov (rastrové, či vektorové vrstvy), alebo vykonávať určité, pre GIS typické analýzy, ale ich hlavnou špecializáciou nie sú geografické informačné systémy (napr. CAD systémy).

• Podľa softvérovej licencie (Štědroň, 2010):

a) Zatvorený, alebo proprietárny softvér

Týmto termínom sú označené GIS softvéry, ktoré sú pre účely práce dodávané len v podobe binárneho kódu bez prístupu ku zdrojovému kódu, a teda bez možnosti úprav užívateľom a prísne limitovaných možností užívania a ďalšieho šírenia. Výhodou tohto typu je stála technická podpora a istota vývoja softvéru, alebo kontrolovaný vývoj programu určeným smerom (Lajčín, 2013). Ako príklad môžeme uviesť väčšinu ESRI (ArcGIS) produktov, IDRISI, Manifold atď.

b) Otvorený, alebo slobodný softvér

Aj keď tieto dva pojmy nie sú identické a rozdiely medzi nimi sú len filozofické, základná myšlienka zdieľania súborov a zdrojového kódu zostáva rovnaká, preto ich možno používať ako synonymá.

Stallman (2002) stanovil štyri základné kritéria slobodného softvéru:

- slobodu spustiť softvér pre akékoľvek účely,
- slobodu skúmať a upravovať program,
- slobodu program kopírovať,
- slobodu vylepšovať softvér a podeliť sa o toto vylepšenie s verejnosťou, pre úžitok celej spoločnosti.

Za výhody slobodných softvérov môžeme pokladať najmä bezpečnosť (na vývoji sa podieľa väčší počet vývojárov, a tým je vyššia šanca nájdenia chyby a jej rýchleho opravenia), flexibilitu (možnosť upravovania programu podľa potreby používateľa), možnosť vyskúšania pred obstaraním, cena a sloboda (Kolektív autorov, 2012). Najznámejšími z rad otvorených softvérov sú GRASS GIS, Quantum GIS, SAGA, DIVA, ILWIS atď.

GRASS GIS – GRASS (Geographic Resources Analysis Support System), je slobodný, multiplatformový, geograficko informačno systémový softvér, určený na analýzu geopriestorových (2-D/3-D rastrových a vektorových) údajov, obrazových záznamov, produkciu grafických a mapových výstupov, priestorové modelovanie a vizualizáciu (Wickert, 2012).

Jeden z najstarších a najoriginálnejších desktop systémov originálne vyvíjaný USA armádou, neskôr podporovaný množstvom vládnych i nevládnych organizácií s viac ako 30 ročným pôsobením (GRASS Development Team, 2015).

1.2 Priestorové údaje

Základnými jednotkami opísanými pomocou priestorových údajov sú tzv. geoobjekty. Rapant (1997) ich definoval ako: „základné priestorové entity ďalej nedeliteľné na jednotky rovnakého typu, popisované priestorovými údajmi. Z geoobjektov je zložené prostredie, v ktorom sa pohybuje človek.“ Pre čo najpresnejšie a najsprávnejšie zobrazovanie a spracovávanie geoobjektov v GIS je nutný ich popis z viacerých hľadísk.

Rapant (2002) definoval základné zložky geoobjektov:

1. Geometrická zložka
2. Tematická zložka
3. Časová zložka
4. Vzťahová zložka

Okrem týchto hlavných zložiek existuje aj tzv. kvalitatívna zložka, ktorá sa ku geoobjektom nevzťahuje priamo, ale dopĺňa ich popis ako taký.

1. Geometrická zložka

Z pohľadu GIS je najdôležitejšou zložkou odlišujúcou GIS od iných informačných systémov (Maguire, 1991). Opisuje jednak tvar objektu (priamy popis geometrických vlastností), ale aj topológiu (priestorové vzťahy s okolím).

Geometrická zložka popisuje:

- **Priestor:** Kvôli definovaniu geometrických vlastností je často využívaný euklidovský priestor, v ktorom poloha môže byť definovaná 2-D (súradnicami x a y), 2.5-D (súradnica z uložená len ako atribút), alebo 3-D (Kusendová, 2003).

- **Určenie polohy v priestore:** Údaje o konkrétnom umiestnení v priestore patria k základným informáciám každého geoobjektu. Proces vytvárania týchto údajov sa nazýva georeferencovanie (Burkholder, 2001).

- **Priestorové vlastnosti geobjektov:** Priestorové vlastnosti geobjektov podľa Komárkovej, Kopáčkovej (2008), možno rozdeliť na dve skupiny:

- Priestorové vlastnosti jednotlivých geobjektov: dĺžka; objem; sklon; tvar; rozloha; orientácia; stred línie, alebo plochy,

- Priestorové vlastnosti skupín geobjektov: počet susedných geobjektov; prednostné smery migrácie v prírode; vzory priestorového rozloženia bodových a plošných geobjektov; prepojenie geobjektov; vzdialenosť medzi geobjektmi; postupnosti geobjektov s rôznymi vlastnosťami.

2. Tematická zložka

Súčasťou tejto zložky je najmä databáza, ktorú chápeme ako zjednotenú množinu účelovo zhromaždených údajov (Escobar et al., 1999). S GIS pohľadu sú najvyužívanejšie relačné databázy tvorené dvojrozmernou tabuľkou s atribútmi o jednotlivých objektoch, kde riadky reprezentujú jednotlivé geobjekty a stĺpce ich popisované vlastnosti (Hofierka, 2003).

3. Časová zložka

Všetky geografické informácie sú vždy viazané k určitému časovému okamžiku. Znalosť časovej stránky geobjektov je veľmi dôležitá pre ich využitie a následnú prácu s nimi (Voženílek, 1998). Časová zložka je využívaná pri rôznych činnostiach, ako napríklad vyhodnocovaní zmien, dopadov človeka na prírodu, pri overovaní podmienok z minulosti alebo pri analýzach s časovou dimenziou atď. (Rapant, 2002).

4. Vzťahová zložka

Každý jeden geobjekt môže byť vo vzájomnom vzťahu s inými geobjektmi, pričom môžu byť týmito vzťahmi značne ovplyvnené (Blake, 2007).

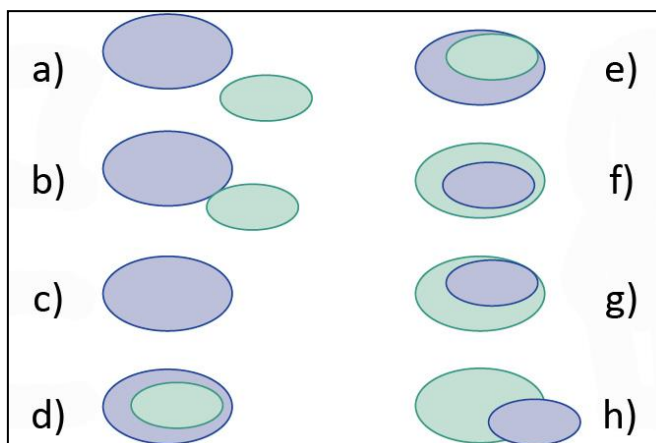
Podľa Estera et al. (2000) rozlišujeme tri základné typy vzťahov medzi geobjektmi:

- **Topologické vzťahy**

V GIS sa topológiou označujú vzťahy medzi jednotlivými geobjektmi. Pre topológiu je typické, že nepracuje so žiadnymi súradnicami objektov a zaoberá sa čisto ich geometrickými a priestorovými vzťahmi, ktoré nemusia byť definované a závislé na žiadnom súradnicovom systéme (Bernhardsen, 1992).

Na základe možných vzťahov medzi objektmi A a B definoval Egenhofer et al., (1989) 8 základných topologických vzťahov (Obr. 1):

- a) A sa nedotýka s B
- b) A sa dotýka B
- c) A sa zhoduje s B
- d) A obsahuje B
- e) A je zakrývané s B
- f) A je vo vnútri B
- g) A zakrýva B
- h) A sa prekrýva s B



Obr. 1 Zobrazenie topologických vzťahov
(Huisman, Rolf, 2009 - upravené)

- **Vzdialenostné vzťahy**

Vzdialenostné vzťahy slúžia na porovnávanie vzdialeností medzi jednotlivými geoobjektmi pomocou danej konštanty a aritmetických operátorov, pričom vzdialenosť medzi týmito geoobjektmi je stanovená ako najkratšia vzdialenosť (Palacio, Sol, González, 2003).

Meranie vzdialenosti v GIS je závislé najmä od použitého systému a metriky. Pri kontinuálnych systémoch sa využíva Euklidovská metrika, pre ktorú platí vzťah:

$$\text{vzdialenosť} = \sqrt{((x_i - x_j)^2 + (y_i - y_j)^2)}$$

Pri práci s diskretnými súradnicovými systémami sa používa „Manhattan“ metrika, ktorá je definovaná ako: $d_{i,j} = (x_i - x_j) + (y_i - y_j)$ (Komárková, Kopáčková, 2008).

- **Smerové vzťahy**

Smerové vzťahy slúžia na opis orientácie geoobjektov v priestore, vzhľadom k ich vzájomnému vzťahu (Santos, Moreira, 2009). Na definovanie smerových vzťahov geoobjektov je potrebné si určiť jeden zdrojový objekt, na základe ktorého sa jednotlivé smerové vzťahy opisujú (Palacio, Sol, González, 2003).

5. Kvalitatívna zložka

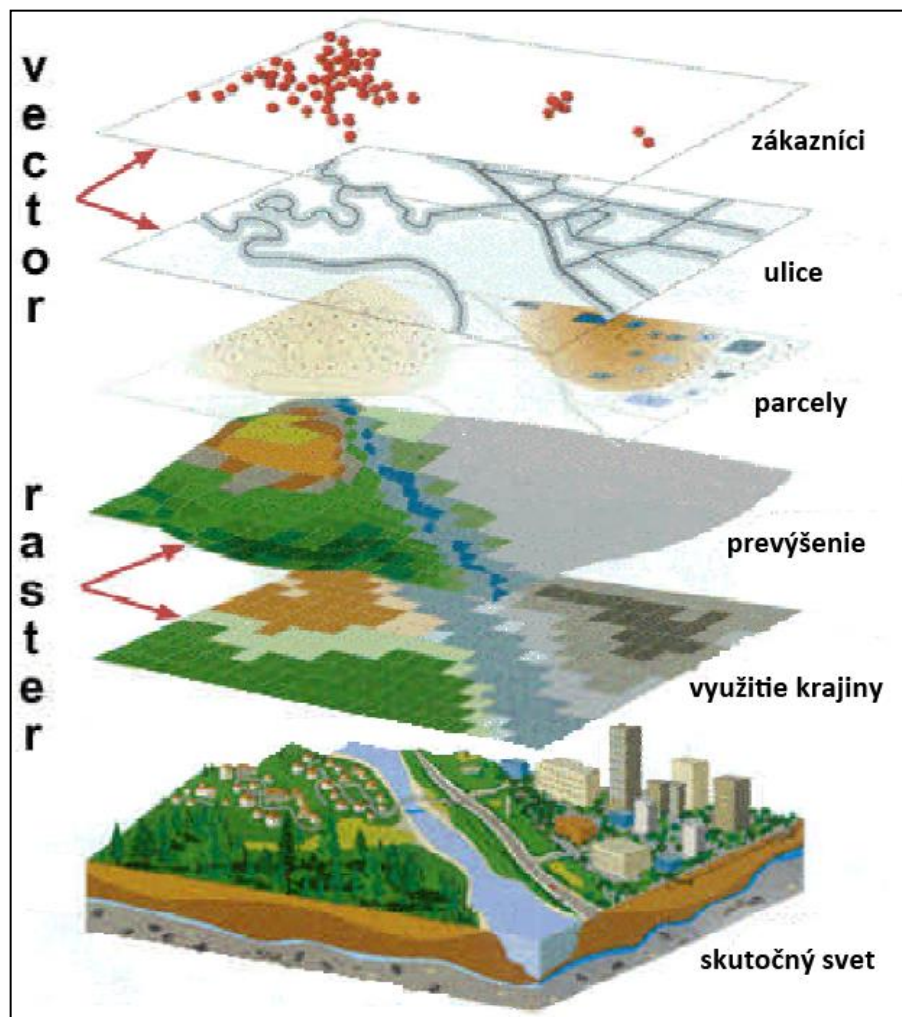
Opis geoobjektu základnými zložkami je nie vždy dostatočný a pre ďalšiu prácu je potrebné tieto údaje doplniť. Kvalitatívna zložka tieto údaje dopĺňa formou metadát. U.S. Department of Transportation (2001) definuje metadáta ako: „Informácie ktoré opisujú alebo charakterizujú údaje. Metadáta sa používajú na dokumentáciu dátových produktov. Metadáta v podstate odpovedajú na otázky kto, čo, kedy, kde, prečo a ako o každom aspekte údajov, ktoré sú zhromažďované alebo zdokumentované.“

1.3 Dátové modelovanie

Pod pojmom dátové modelovanie podľa Bernhardsena (1992) rozumieme proces interpretácie reálneho sveta, pri ktorom zdôrazňujeme podstatnejšie časti a nepodstatné eliminujeme – avšak s ohľadom na cieľ ktorý ma toto modelovanie splniť.

Komárková, Kopáčková (2008) členia dátové modely na:

- **Klasické** (Obr. 2)
 - vektorový
 - rastrový
 - hybridný (spoločné a jednotné zapracovanie vektorových aj rastrových údajov.)
- **Objektovo orientované:** Tradične sú kódy a údaje jednotlivých objektov uchovávané v oddelených častiach. Pri objektovo orientovaných modeloch sú zlúčené do jedného neoddeliteľného celku – objektu. Tieto typy modelov umožňujú prirodzenejšiu reprezentáciu zložitejších dátových štruktúr, čím sa toto spracovanie blíži viac k realite. (U.S. Department of Transportation, 2001).



Obr. 2 Konceptia rastrových a vektorových vrstiev (Escobar et al., 1999 - upravené)

1.3.1 Vektory

Vektorové dátové modely sú jedným z prvých a najrozšírenejších modelov používaných v súčasných GIS. Ich hlavnou výhodou, okrem vysokej presnosti, je oddelené uloženie geometrickej a tematickej zložky popisu geoobjektu, pričom väzba medzi týmito zložkami je zabezpečená pomocou unikátneho identifikátora pre každý geoobjekt (Escobar et al., 1999). Vektorové reprezentácie zachytávajú objekty reálneho sveta pomocou základných geometrických typov (Longley et al., 2005).

Břehovský, Jedlička (2005) delia jednotlivé geometrické typy na:

- **Bod** – vektor o nulovej dĺžke, ktorý z geometrického pohľadu nemá žiaden rozmer. Je definovaný súradnicami (x , y , prípadne z) a jeho topologickou reprezentáciou je uzol (Hrubý, 2006),

- **Línia** – je sériou usporiadaných bodov. Pri líniách rozlišujeme body začiatkové a koncové označované ako uzly a body vnútorné označované ako vrcholy. Vzhľadom k tomu, že línia má len jeden rozmer (dĺžku), patrí k 1-D objektom. Jej topologickým ekvivalentom je hrana,

- **Plocha** – z geometrického hľadiska môžeme plochu opísať ako uzavretú líniu či reťazec línií, ktorých prvý aj posledný uzol sú rovnaké. Patrí k 2-D objektom,

- **Povrch** – typ plochy, ktorá má v každom svojom bode priradené určité hodnoty (napr. nadmorskú výšku). Je označovaný ako tzv. 2.5-D objekt,

- **Objem** – Patrí k 3-D objektom, ale kvôli svojej jeho vysokej náročnosti na výpočet sa v praxi moc nevyužíva.

Keďže počítač len na základe jednoduchých geometrických typov nedokáže zistiť ktorá línia susedí s ktorým polygónom, často sa k týmto dátovým modelom zavádza aj topológia. Existuje viacero vektorových dátových modelov, ktoré môžeme podľa spôsobu ukladania geometrie geoobjektov rozdeliť do dvoch skupín (Rapant, 2002):

- **Nespojité**

Jedným z najstarších a najjednoduchších nespojitých modelov je tzv. špagetový model. Špagetový model vychádza z digitalizácie máp, pri ktorej je každý geoobjekt kódovaný samostatne vo vektorovej forme a reprezentovaný jedným logickým záznamom bez vytvárania vzťahov s okolitými objektmi. Jeho hlavnou nevýhodou je spôsob ukladania susediacich polygónov, pričom spoločná línia je ukladaná dvakrát. Kvôli tomu tento model nie je schopný poskytnúť informácie o vzťahoch medzi jednotlivými geoobjektmi. Línie sa tu môžu ľubovoľne krížiť, kvôli čomu je tento model pre väčšinu

priestorových analýz nevhodný. Svoje miesto si našiel najmä v počítačovej grafike a digitálnej kartografii (Rapant, 2002).

- **Topologické**

Medzi najčastejšie používané modely patria základný topologický model a hierarchický model. Hlavnou výhodou základného topologického modelu oproti špagetovému je možnosť analýzy väzieb a vzťahov medzi objektmi, kde sú tieto topologické informácie uložené v databázach. Vďaka tomu je určovanie a analyzovanie susedských vzťahov geoobjektov oveľa ľahšie (Hofierka, 2003).

Hierarchické dátové modely sú svojim spôsobom ukladania údajov veľmi blízke reálnemu svetu. Avšak v GIS sa tieto modely moc neujali, najmä kvôli pevne zabudovaným údajom do logickej schránky dátovej štruktúry, čím sa tieto údaje veľmi ťažko menia či dopĺňajú (U.S. Department of Transportation, 2001).

1.3.2 Rastre

Rastrové údaje v GIS chápeme ako matice diskretných buniek, ktoré reprezentujú objekty na, nad alebo pod zemským povrchom (Sherman et al., 2007).

Základným stavebným prvkom rastrových modelov je bunka. Tieto bunky sa ďalej usporiadávajú do tzv. mozaiky, pričom každá z buniek obsahuje hodnoty zastupujúce skúmanú vlastnosť zemského povrchu (Eastman et al., 1993). V 2-D digitálnych obrazov je bunka označovaná ako pixel (*angl. picture element*) a v 3-D reprezentáciách je ekvivalentom pojem voxel (*angl. volume element*) (Huisman, Rolf, 2009).

Dátovým typom rastrových buniek môže byť buď určitý číselný typ, ako *integer*, či *float*, ktorý vyjadruje hodnotu popisovaného priestorového procesu, alebo kód ktorý skúmanú oblasť kategorizuje na základe spoločných vlastností (Longley et al., 2005). V mozaikovom rastrovom modeli môže nastať aj situácia, kedy bunka nemá priradenú žiadnu hodnotu, tzv. NoData, ktorá sa pri analýzach rieši zavedením špeciálnej prázdnej hodnoty (Hrubý, 2006).

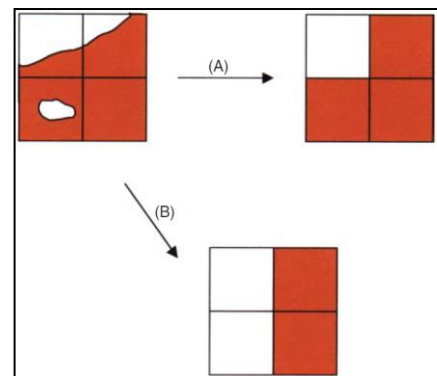
Požiadavky na bunku rastru podľa Komárkovej, Kopáčkovej (2008):

- bunka by mala byť nekonečne opakovateľná v rovine – kvôli možnosti pokryť ľubovoľnú veľkú oblasť bez zvyšku,
- bunka by mala byť nekonečne rekurzívne deliteľná na menšie bunky rovnakého tvaru – pre použitie hierarchickej dátovej štruktúry ukladania údajov.

Keďže každej bunke je priradená len jedna hodnota, ktorá nesie jednoznačnú informáciu o reprezentovanom jave pokrývajúca určité územie, je veľmi dôležité vybrať veľkosť bunky, ktorá by modelovaný jav dobre reprezentovala. Veľkosť bunky definuje rozlíšenie rastra. Pre čo najlepšie reprezentovanie je nutné zachytiť jav aj v miestach s jeho prudkými zmenami. Problém nastáva pri redundancii údajov, keď daný jav zasahuje do bunky len čiastočne a je potrebné vybrať kritérium pre reprezentovanie príslušnou hodnotou v bunke. (Hofierka, 2003).

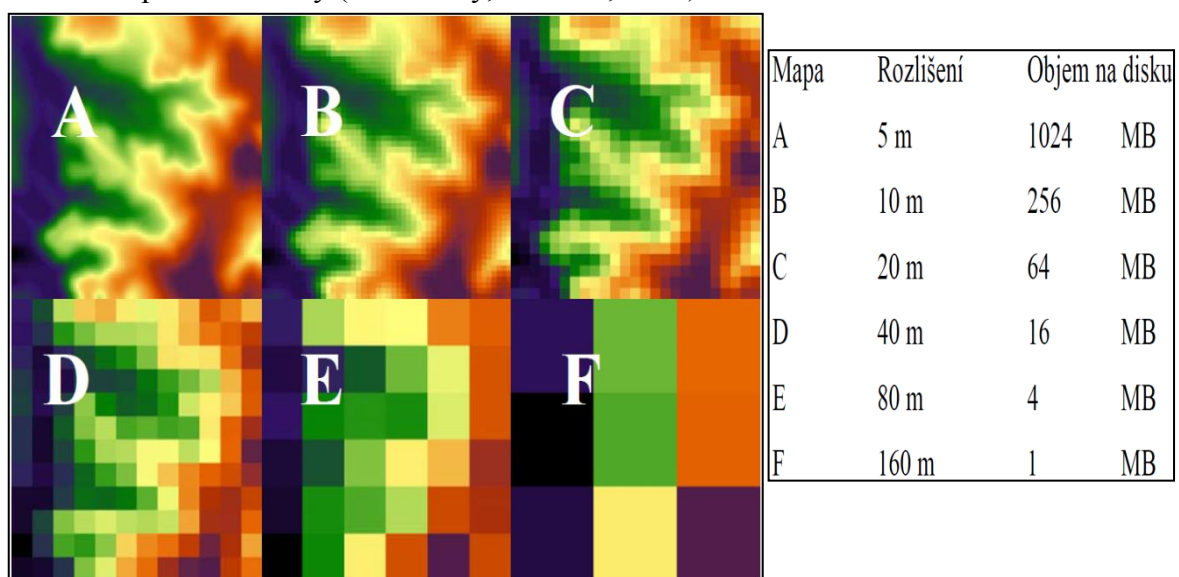
Hrubý (2006) rozlišuje 3 základné prístupy na reprezentovanie hodnoty bunky (Obr. 3):

1. Metóda centroidov – bunka bude reprezentovaná podľa hodnoty, ktorá je najbližšia jej stredu,
2. Metóda dominantného typu – bunku reprezentuje hodnota s prevládajúcim výskytom,
3. Metóda najdôležitejšej kategórie – užívateľ si sám navolí významnosť jednotlivých kategórií.



Obr. 3 Metóda dominantného typu (A) a metóda centroidov (B) (Longley et al., 2005)

Vďaka implicitne definovanými vzťahmi medzi jednotlivými bunkami je rastrový model z hľadiska topológie oveľa jednoduchší než vektorový. Reprezentácia rastrových modelov je typická svojou veľkou a náročnosťou na množstvo uložených informácií. Veľkosť jednotlivých údajov je individuálna a závisí od rozlohy záujmovej oblasti a použitom rozlíšení (Obr. 4). Vzhľadom k tomu sa na ukladanie týchto údajov využívajú rôzne kompresné metódy (Břehovský, Jedlička, 2005).



Obr. 4 Miera ovplyvnenia kvality a objemovej náročnosti vzhľadom na zmenu rozlíšenia (Krása et al., 2006)

1.4 Zber údajov

Existuje viacero spôsobov zberu a zdrojov údajov, ktoré možno rozdeliť na dva hlavné zdroje (Jackson, Woodsford 1991).

- **Primárne zdroje/údaje:** Metódy pri ktorých vznikajú nové, predtým nejestvujúce údaje, a to najmä priamym meraním v teréne. Získavajú sa tak najpresnejšie a najkvalitnejšie údaje. Avšak tieto metódy sú veľmi pracné a finančne a časovo náročné (Hrubý, 2006).

Medzi najčastejšie využívané primárne zdroje patrí (Burkholder, 2001; Tuček, 2002):

- geodetické zameriavanie: najpresnejší a najdokonalejší spôsob získavania údajov založený na meraní uhlov a vzdialeností miest z bodov so známou polohou,
- fotogrammetria a fotointerpretácia: vedné odbory zaoberajúci sa rekonštrukciou tvaru, veľkosti, polohy a interpretáciou predmetov na fotogrametrických snímkach,
- diaľkový prieskum zeme: zber údajov bez priameho kontaktu, požitím najmä satelitných a digitálnych snímkov a ich spracovanie a interpretácia na potrebné informácie
- GPS: využívaný na získavanie polohy referenčných bodov a jednoduchú lokalizáciu objektov.

- **Sekundárne zdroje/údaje:** Ide o údaje, ktoré boli v minulosti primárnymi zdrojmi a pre ich terajšie využitie je potrebné ďalšie spracovanie. Pri tomto type informácií treba brať do úvahy chyby získané už počas prvého spracovania údajov a tak nie sú natoľko presné ako metódy primárneho zberu (Břehovský, Jedlička, 2005).

Longley et al. (2005) delí sekundárne zdroje podľa formy na:

- analógové – texty, mapy, registre, grafy, tabuľky, zoznamy, kartogramy atď.,
- digitálne – digitálne mapy, databázy, snímky z družíc atď.

Podľa Huismana a Rolfa (2009) poznáme niekoľko metód získavania sekundárnych zdrojov:

- manuálne cez klávesnicu: veľmi pracné, zdĺhavé a na spracovanie väčšieho počtu údajov prakticky nepoužiteľné,
- manuálna digitalizácia: metóda snímania podkladu pomocou zameriavacieho zariadenia a ručného zadávania identifikátorov jednotlivých objektov,
- scanovanie a vektorizácia: prevod údajov z analógovej do digitálnej formy scanovaním a následný proces identifikácie jednotlivých objektov.

1.5 Priestorové analýzy

Možnosť vykonávať priestorové analýzy je jednou z najdôležitejších a kľúčových častí všetkých GIS technológií, a taktiež tým, čím sa GIS odlišujú od ostatných informačných systémov (Theobald, 2007).

Horák (2002) definuje priestorové analýzy ako: „Súbor techník pre analýzu a modelovanie lokalizovaných objektov, kde výsledky analýz závisia od priestorového usporiadania týchto objektov a ich vlastností.“ Inými slovami, skúmaním priestorových analýz zisťujeme všetky súvislosti medzi rôznymi aspektmi priestorových modelov, ktoré nemusia byť na prvý pohľad zrejmé, pričom získavame predtým neexistujúce informácie.

Analýzy v GIS sú komplexné metódy, ktoré podľa Komárkovej, Kopáčkovej (2008) berú do úvahy geometriu (veľkosť, polohu, tvar objektov), štatistické chovanie a vlastnosti objektov, povrch a digitálne modely reliéfu.

Hrubý (2006) rozdelil najdôležitejšie analytické funkcie GIS do nasledujúcich 7 kategórii:

1. Dopytovanie na databázy

Pod databázovým dopytom chápeme vyberanie údajov, ktoré zodpovedajú určitému kritériu alebo špecifickej podmienke (Eastman et al., 1993).

Tuček (1998) uvádza tieto tri hlavné komponenty databázového dopytovania:

- spresnenie údajov, ktorých sa dopyt týka,
- zavedenie podmienok, ktorým musia údaje vyhovovať,
- pokyny, čo sa má na vybraných údajov vykonať.

Databázový dopyt v GIS môžeme podľa Huismana a Rolfa (2009) rozdeliť na:

- **Atribútový** – na vyhľadávanie nevyužíva priestorové údaje o geoobjektoch, ale pracuje iba s popisnými informáciami. Existuje viacero variantov pre identifikáciu geoobjektov. Často využívané je vyhľadávanie na základe mena, označenia, ID kódu atď., alebo častejšie na základe identifikovaní všetkých objektov spĺňujúcich intervalové a logické podmienky jedného, alebo viacerých atribútov (Obr. 5).

Pre databázový dopyt týkajúci sa viacerých atribútov sa používajú matematické (<, >, =, <=, >=, <> (!=)) a logické (AND, OR, NOT) operátory s využívaním pravidiel Booleovskej logiky (pravda=1, nepravda=0) (Komárková, Kopáčková, 2008).

GRASS GIS Attribute Table Manager - <KATASTRE_10000@obr>

1 / Table KATASTRE_10000

Attribute data - right-click to edit/manage records

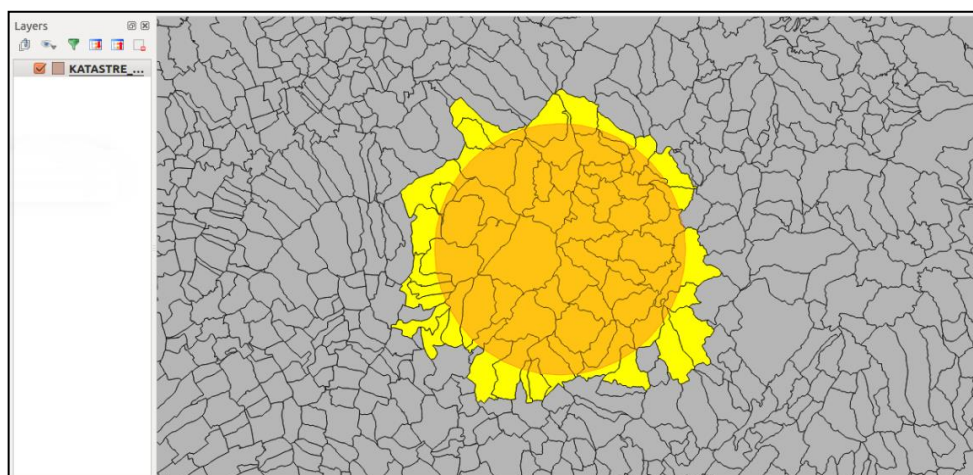
cat	NAZOBCE	NAZOKRES	ROZLOHA_HA
327	Tekovské Lužany	Levice	3220.76
348	Želiezovce	Levice	3720.06
355	Farná	Levice	3259.04
359	Veľké Ludince	Levice	3184.16
893	Veľké Zálužie	Nitra	3205.6

Query

```
SELECT * FROM KATASTRE_10000 WHERE (NAZOKRES = 'Nitra' OR NAZOKRES = 'Levice') AND (ROZLOHA_HA >= 3000 AND ROZLOHA_HA <= 4000)
```

Obr. 5 Atribútový dopyt pre obce v okrese Nitra a Levice s rozlohou od 3000 do 4000 ha

- **Priestorový** – môže byť uskutočnený dvoma spôsobmi. Prvý spočíva vybraním geoobjektu na základe jeho súradníc a to buď ich ručným zadaným, alebo interaktívne pomocou myši. Druhým spôsobom je vyhľadávanie na základe rôznych geometrických tvarov (kocky, kružnice, línie atď.) (Huisman, Rolf, 2009) (Obr. 6).



Obr. 6 Rádiusový priestorový dopyt v prostredí QGIS

- **Kombinovaný** – použitie kombinovaného dopytu umožňuje používať topologické prekrytie (Břehovský, Jedlička, 2005). Pod pojmom topologické prekrytie chápeme dopytovanie sa na dvoch a viac vrstvách zároveň. Toto prekrytie je využívané ako vo vektorovej, tak aj v rastrovej reprezentácii. Vektorová reprezentácia dovoľuje oddelene uložiť geometriu a popisnú časť objektov, vďaka čomu je možné vytvárať nové vrstvy s kombinovanými výslednými atribútmi.

Pri rastrovej reprezentácii je možné toto prekrytie vykonať (Hrubý, 2006):

- reklasifikáciou: ide o výber buniek spĺňajúcich určenú podmienku kladenú na ich atribúty,
- mapovou algebrou.

2. Mapová algebra

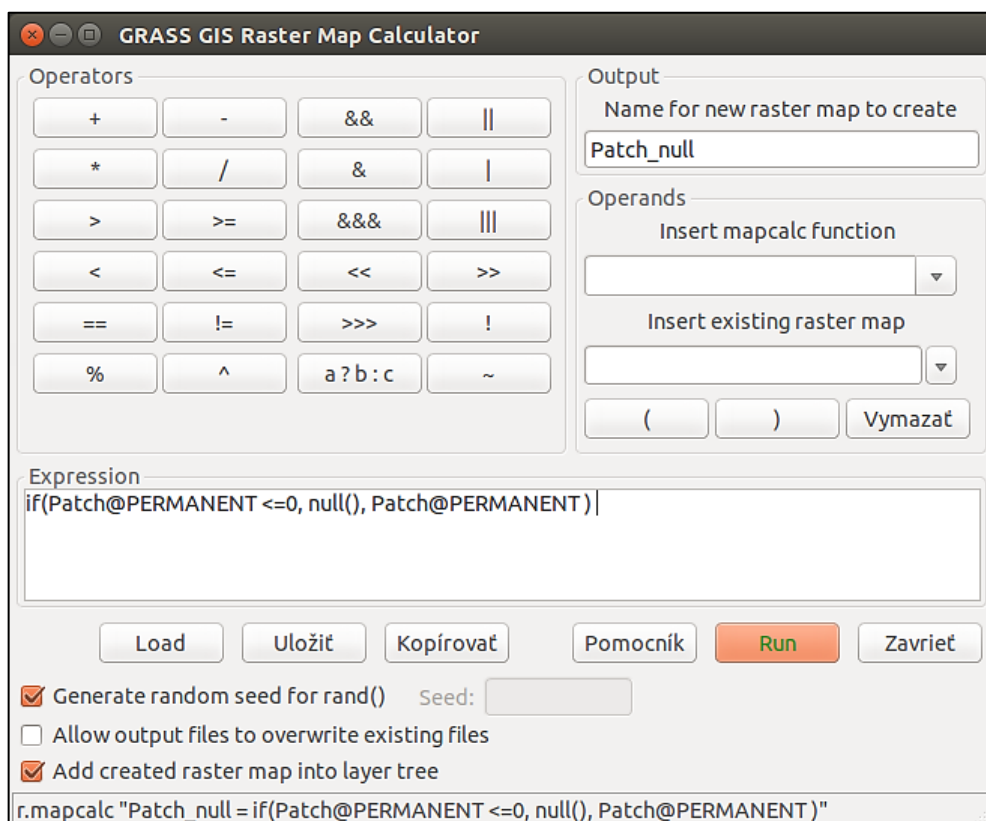
Keďže rastrovú mapu možno chápať ako maticu čísel, kde každej hodnote atribútu priradíme určitý význam, môžeme tieto rastre vzájomne kombinovať pomocou rôznych matematických operácií (Câmara et al., 2005). Mapová algebra využíva objekty, činnosti a kvalifikátory činností (Hrubý, 2006).

Objekty chápeme buď ako vstupné hodnoty, alebo ako spôsob uloženia informácií. Môžu to byť tabuľky, celé rastre, konštanty atď.

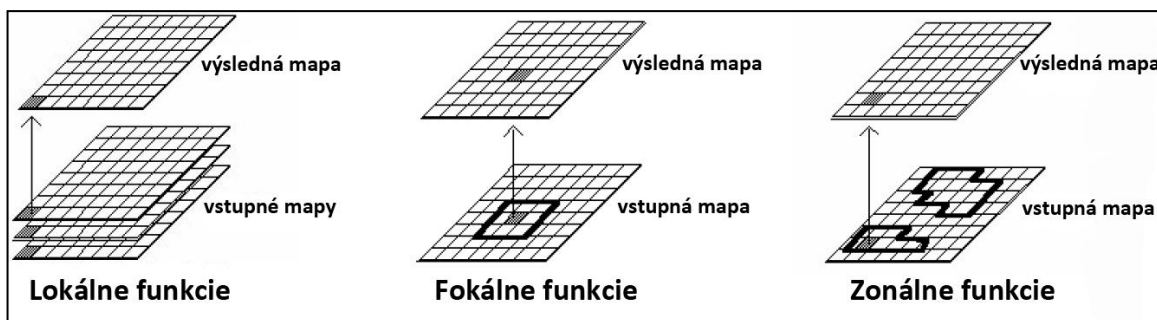
Činnosti sú príkazy a operácie vykonávané na objektoch. Rozdelenie operátorov podľa Komárkovej, Kopáčkovej (2008):

- aritmetické (mod; +; -; *; /),
- Booleanovské (NOT; AND; OR; XOR),
- relačné (<; >; =; ^=),
- kombinatorické (CAND; COR; CXOR),
- logické (DIFF; IN; OVER),
- bitové presuny (^ ^; >>; <<; &&; !!; ||).

Kvalifikátory činností riadia vykonávanie jednotlivých činností (cykly, konštrukcie, podmienky atď.) (Břehovský, Jedlička, 2005) (Obr. 7).



Obr. 7 Ukážka mapovej algebry



Obr. 8 Znáročenie funkcií mapovej algebry (Câmara et al., 2005 - upravené)

Funkcie mapovej algebry je možné podľa Tomlina (2012) (Obr. 8) rozdeliť na:

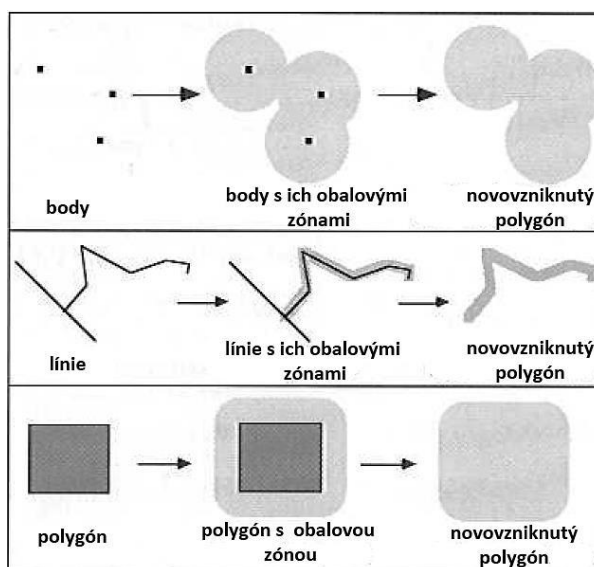
- **lokálne** – nové hodnoty vznikajú prekryvom individuálnej bunky jednej alebo viacerých vrstiev,
- **fokálne** – hodnoty vznikajú z definovaného okolia bunky,
- **zonálne** – nové hodnoty vznikajú zo špecifickej oblasti definovanej v inej vrstve,ň
- **globálne** – počítanie prebieha zo všetkými bunkami vrstvy.

3. Vzďialenostné analýzy

Meranie vzdialenosti patrí k základným operáciám každého GIS softvéru, a v podstate každej mapy. Ako bolo spomenuté vyššie (kapitola 1.2 - Vzďialenostné vzťahy), na meranie vzdialenosti sa používa buď Euklidovská (pri vektorovej reprezentácii) alebo Manhattan metrika (pri rastrovej reprezentácii). Vzďialenostná analýza môže mať v GIS viacero významov.

Aspekty vzdialenostných analýz podľa Hrubého (2006):

- **Obalová zóna** – najpoužívanější vzdialenostná analýza spočívajúca vo vytvorení novej vrstvy – obalu v predom určenej vzdialenosti okolo jednotlivých bodov, línií, alebo polygónov (Obr. 9) (Patra, 2013).



Obr. 9 Tvorba obalových zón (Břehovský, Jedlička, 2005 - upravené)

- **Frikčné povrchy** – pri bežných vzdialenostných analýzach sa vôbec nepočíta s okolnými vplyvmi a všetko je merané vzdušnou čiarou za ideálnych podmienok, čo vôbec nezodpovedá reálnemu svetu. Pri frikčných povrchoch je každej bunke priradená určitá “cena“, čím sa vytvorí tzv. nákladový vzdialenostný povrch, ktorý sa viac podobá skutočnosti (Eastman et al., 1993). Faktory ovplyvňujúce frikčné povrchy podľa Komárkovej, Kopáčkovej (2008):

- faktor terénneho reliéfu (výškové údaje DMT),
- vertikálny faktor (sklon, koncentrácia častíc atď.),
- horizontálny faktor (smer presunu, vplyv vodných prúdov atď.).

- **Funkcie šírenia a prúdenia** – funkcia šírenia je výpočet matice kumulatívnej vzdialenosti všetkých bodov od začiatočného po konečný bod založený na frikčnom povrchu. Funkcia prúdenia je opačnou funkciou, pri ktorej výsledkom je najrýchlejšia cesta z cieľového bodu do bodu s nulovým nákladovým potenciálom (Hrubý, 2006).

4. Analýzy terénu

Pri analýze modelov terénu väčšinou hovoríme o analýzach s použitím výšok terénu. Voženílek (1998) ich charakterizoval ako: Nástroje na vytváranie štruktúr reprezentujúce spojité povrchy, prebiehajúce vo vektorovom i rastrovom formáte. V 3-D priestore sú simulované povrchy na výpočty morfometrických charakteristík, odtokových pomerov, viditeľnosti a pod.

Medzi základné morfometrické operácie podľa Mitášovej a Hofierku (1993) patrí:

- **uhol sklonu reliéfu v smere spádovej krivky**: najčastejšie používaný morfometrický ukazovateľ reliéfu slúžiaci na vytváranie izoklín vybraných hodnôt, ktoré ohraničujú areály s daným intervalom sklonov,

- **orientácia reliéfu voči svetovým stranám**: vyjadruje sa izotangentami, ktoré spájajú body s rovnakou orientáciou reliéfu,

- **normálová krivosť v smere spádovej krivky**: využitie pri určovaní tendencie pohybu –zrýchľovanie, spomaľovanie vody a materiálov po svahu,

- **normálová krivosť v smere dotyčnice k vrstevnici**: hlavný ukazovateľ smeru odtoku a integrácie vody a materiálu po svahoch (Miklós, Špinerová, 2010).

5. Analýza sietí

Analýzy sietí môžeme chápať ako vzdialenostné analýzy s váženou (nákladovou/frikčnou) vzdialenosťou s rozdielom možnosti použitia len vo vektorových modeloch. Pred použitím siet'ových analýz je potrebné si vytvoriť sieť.

Tvorba siete spočíva vo (Břehovský, Jedlička, 2005):

- získaní líniovej vrstvy, na ktorej budú analýzy uskutočňované,
- zavedenia pravidiel uložených do atribútových tabuliek. Ide o uzlové pravidlá (určujúce smer pohybu cez uzol) a hranové pravidlá (určujúce smer a rýchlosť pohybu po hrane),
- pridanie informácii pre výstupy analýz (adresy, mená ulíc atď.).

Využitie: vyhľadanie trasy, modelovanie zaťaženia sietí, alokácia zdrojov atď.

6. Analýza obrazov

V rámci analýz obrazov ide predovšetkým o spracovávanie leteckých fotografií a fotografií z diaľkového prieskumu Zeme. Môžeme ich chápať ako spôsob zisťovania a merania reality (Tuček, 1998).

Podľa Komárkovej, Kopáčkovej (2008) sa analýza obrazov zameriava na tri hlavné aktivity:

- korekcia: opravovanie chýb v obrazoch (šumy, atmosféra, chyby snímačov),
- zlepšenie: metódy na vylepšenie obrazu z hľadiska možnosti ďalšieho spracovania (zmena jasů, farby, kontrastu atď.),
- klasifikácia: metódy umožňujúce zapracovanému obrazu identifikovať homogénne oblasti.

7. Simulačné procesy

Ide o napojenie GIS na rozsiahle výpočty kombinujúce priestorové modely a procesy. Spojením možnosti simulácie modelov spolu s priestorovými technológiami vznikajú jedinečné systémy schopné využívať silné stránky z oboch oblastí. Simulačné modely sú sofistikované nástroje na charakterizovanie a pochopenie prírodných vzorcov a procesov a odhadovanie účinkov prírodných zmien na lokálnej, regionálnej, či globálnej úrovni.

Patria sem atmosferické, hydrologické, či krajinárske simulácie a rozličné modely pohromových situácií (Hrubý, 2006; Steyaert, Goodchild, 1993; Silva, Eglesse, 2000).

2 METODIKA PRÁCE

2.1 Opis vzniku skriptov

Keďže cieľom práce bolo vytvorenie softvérových nástrojov s využitím v rôznych krajino-ekologických oblastiach, vytvorili sme viacero skriptov s rozličným zameraním a možnosťou využitia v mnohých činnostiach a odvetviach ekológie a environmentalistiky. Vzhľadom k open-sourcovej logike a faktu, že práca so skriptami nie je nijak obmedzená, ich využitie závisí hlavne od zručnosti a aktuálnych potrieb užívateľa. Jednotlivé skripty sme však rozdelili do štyroch základných kategórií na základe primárnych možnosti využitia:

a) **Výpočet reálnej rozlohy:** Skript vznikol ako odpoveď na otázku „*Aký je plošný rozdiel medzi rozlohou počítanou klasickými GIS nástrojmi a skutočnou 3-D rozlohou územia ?*“

Z dôvodu členitosti záujmových území bol skript využitý napríklad pri rátaní a porovnávaní rozlôh jednotlivých osád z minulosti a súčasnosti v diplomovej práci „*Vývoj rozptýleného osídlenia v mestách Krásno nad Kysucou, Turzovka a Čadca*“ od Bc. Dominiky Kaisovej (Kaisová, 2015).

b) **DBMS a tvorba priestorových údajov:** Tieto skripty vznikli v spolupráci s Mgr. Adamom Konečným, Ph.D. z ústavu botaniky a zoológie na Masarykovej univerzite v Brne. Na základe žiadosti o prácu s databázou v špecifickej forme boli vytvorené tri skripty určené na analýzu a tvorbu databáz a priestorových údajov. Využitie si však našli aj v tíme doc. Mgr. Ivana Baláža, PhD., z katedry ekológie a environmentalistiky na Univerzite Konštantína Filozofa (UKF) v Nitre pri georeferencovaní pascí odchytu drobných zemných cicavcov kvadratickou metódou.

c) **Výpočet viditeľnosti pomocou paralelných výpočtov:** Podnet vytvorenia týchto skriptov vznikol počas prezentácie - „*Tatry očami geografa*“ od doc. RNDr. Jána Laciku, CSc. z katedry geografie a regionálneho rozvoja na UKF v Nitre, ktorý sa vyjadril, že by bolo dobré mať k dispozícii mapy viditeľnosti z tatranských vrchov, ktoré v súčasnosti chýbajú, a to najmä kvôli vysokej náročnosti na výpočty v prostredí GIS. Skripty boli využité na výpočet viditeľnosti z vybraných slovenských vrchov a hradov.

d) **Simulácia zaplavenia územia:** Skript vznikol v spolupráci s tímom doc. Mgr. Ivana Baláža, PhD., ako nástroj na simulovanie inundačných území pri výskume priestorových interakcií piskorov v lesnom ekosystéme Tatier. Upravené výstupy boli použité na kongrese Zoológia, 2014 v Prešove (Baláž, et al. 2014a) a na medzinárodnej

konferencii Rodent Biology v Lisabone (Baláž, et al. 2014b). Pre potreby práce: “Revitalizácia súčasného stavu MVN” od Bc. Patrika Belianského, bol skript využitý na simulovania zaplaveného územia a zatopených objektov vodnej nádrže vo Veľkých Uherciach (Belianský, 2015).

2.2 Tvorba skriptov

Kvôli výhodám ako variabilita, rýchlosť, stabilita, multitasking, efektívne využívanie virtuálnej pamäte či úplná otvorenosť a sloboda, bol ako pracovné prostredie zvolený UNIX operačný systém, a to konkrétne Linux Ubuntu (14.04.2 LTS – Trusty Tahr).

Vďaka úplnej kompatibilite s GRASS GIS a možnosti využívať všetky vyššie spomenuté výhody zvoleného operačného systému, bol pre tvorbu skriptov použitý BASH. Ide o mocný shell interpretovaný skriptovací jazyk, ktorý je súčasťou skoro každého UNIX operačného systému. Jeho najväčšou výhodou je schopnosť využívať všetky nástroje a možnosti celého systému. Okrem BASH je v skriptoch využitý aj programovací jazyk AWK, a to najmä na prácu s textom.

Pre vytvorenie čo najprijateľnejšieho užívateľského prostredia, je v skriptoch použitý nástroj Zenity, ktorý umožňuje interaktívnu komunikáciu s užívateľom pomocou dialógových okien. Ďalším z externých nástrojov, ktoré nemusia patriť k základnej výbave každého shell-u, skripty využívajú aj vykreslovací program Gnuplot, ktorý vytvára grafy funkcií či výsledných dát.

GRASS GIS, ako jeden z mála plne programovateľných open-sourcových softvérov, umožňuje prácu čisto cez textový režim pomocou príkazov a predprogramovaných nástrojov.

Tab. 1 Použité GRASS GIS nástroje

Zobrazovanie; databáza; všeobecné	Raster	Vektor
<i>d.barscale</i>	<i>r.colors</i>	<i>v.buffer</i>
<i>d.legend</i>	<i>r.in.gdal</i>	<i>v.extract</i>
<i>d.mon</i>	<i>r.lake</i>	<i>v.in.ascii</i>
<i>d.rast</i>	<i>r.mapcalc</i>	<i>v.info</i>
<i>d.vect</i>	<i>r.mask</i>	<i>v.in.ogr</i>
<i>db.out.ogr</i>	<i>r.out.gdal</i>	<i>v.out.ogr</i>
<i>db.select</i>	<i>r.patch</i>	<i>v.overlay</i>
<i>g.gisenv</i>	<i>r.recode</i>	<i>v.report</i>
<i>g.mapset</i>	<i>r.slope.aspect</i>	<i>v.split</i>
<i>g.proj</i>	<i>r.stats</i>	<i>v.to.points</i>
<i>g.region</i>	<i>r.to.vect</i>	<i>v.to.rast</i>
<i>g.remove</i>	<i>r.univar</i>	
<i>g.rename</i>	<i>r.viewshed</i>	
<i>g.version</i>		
<i>GRASS_BATCH_JOB</i>		

Tabuľka 1 obsahuje 39 hlavných GRASS GIS nástrojov využitých v jednotlivých skriptoch. Vzhľadom k tomu, že v samotných skriptoch je použitých vyše 200 variácií týchto nástrojov, je ponechaný ich základný tvar bez parametrov (*flags*).

2.3 Štruktúra skriptov

Pri písaní skriptov bola použitá určitá forma uniformity, a tým pádom sa skripty vyznačujú istou spoločnou štruktúrou.

Pre čo najväčšiu kompatibilitu a možnosť využitia v rôznych systémoch je vo všetkých skriptoch pre zadefinovanie cesty k interpreteru využitý `#!/usr/bin/env bash` shebang.

Každý skript obsahuje hlavičku s najdôležitejšími informáciami. Prvou je samotný názov skriptu. Ak je skript vyvolávaný ako podprogram iného skriptu (napríklad pri výpočte viditeľnosti), je potrebné, aby jeho názov ostal nezmenený. Tento záznam užívateľa informuje o originálnom názve skriptu pre prípad náhodného premenovania. Druhý záznam nesie mená autorov a e-mailový kontakt v prípade hlásenia chýb či návrhov na vylepšenia. Ďalším riadkom je dátum vytvorenia daného skriptu. Keďže bolo vytvorených viacej verzií skriptov (pridané či odobrané niektoré moduly, funkcie, atď.), štvrtý záznam informuje o čísle verzie a pridaných vylepšeniach či odobraných častiach. Piaty záznam informuje o BASH verzii, v ktorej bol skript vytvorený. V šiestom zázname je zapísaná GRASS verzia, v ktorej bol skript vytvorený a testovaný. Vzhľadom k tomu, že začiatkom tohto roku (20.02.2015) vyšla nová stabilná verzia GRASS GIS 7.0.0, v ktorej syntax väčšiny nástrojov bol pozmenený a samotný softvér prešiel zásadnými zmenami, boli všetky skripty aktualizované na túto verziu. Ďalšie dva záznamy obsahujú vstupy a výstupy. Posledným je stručný popis skriptu.

Jednotlivé logické časti skriptov sú navzájom oddelené a opísané formou komentárov (mriežka - #), pričom samotné gro skriptu je vždy označené tromi mriežkami (###), väčšie úseky, ako vytvorenie tabuľky, výsledného grafu atď. dvoma mriežkami (##) a niektoré dôležitejšie časti sú označené jednou mriežkou, pri ktorej sa komentár viaže len k najbližším riadkom.

Niektoré skripty obsahujú funkciu "help". Ak je skript obsahujúci túto funkciu spustený s parametrom `--help` (Nazov_skriptu.sh `--help`), tak sa užívateľovi do terminálu zobrazia informácie o možnostiach a podmienkach pre správne fungovanie daného skriptu.

Pre kontrolu správneho priebehu výpočtu obsahujú skripty niekoľko druhov ukončení a chybových okien (0- bezchybné ukončenie; 1- predčasné chybové ukončenie; 2- predčasné, bezchybné ukončenie v rámci modulu)

2.4 Testovanie skriptov

Väčšina testov uvedených v práci, prebiehalo na štvorjadrovom (2 jadrá, 4 thready) Intel Core i3 350M (2,26GHz) notebooku s 4 GB DDR3 (1066 MHz) pamäťou. Avšak niektoré z testov výpočtu viditeľnosti boli kvôli svojej časovej a hardvérovej náročnosti uskutočnené v GIS učebni na fakulte prírodných vied, kde prebiehali niekoľkodňové výpočty na 11-tich štvorjadrových počítačoch (Obr. 10). Testovanie prebiehalo aj na osemjadrových počítačoch v počítačovej učebni na katedre informatiky, v spolupráci s PaedDr. Petrom Švecom, Ph.D.



Obr. 10 Test viditeľnosti pomocou paralelných výpočtov v GIS učebni – FPV UKF

Pri testoch bolo použitých viacero typov vstupných dát, ktoré boli upravené do potrebnej podoby (vyhladenie, zmena rozlíšenia, korektúra geometrických chýb, úprava atribútov, atď.).

Z pohľadu súradnicových systémov je výber čisto na užívateľovi. Jedinou podmienkou, ktorá pre správne fungovanie väčšiny skriptov musí byť splnená je, že mapové jednotky daného systému sú metre. Niektoré skripty obsahujú podmienku, ktorá mapové jednotky kontroluje a skript ukončí, ak projekcia nie je v metroch. Výpočty v testoch prebiehali v súradnicovom systéme S-JTSK (Ferro) / Krovak (EPSG:2065).

Použité rastrové vrstvy:

- digitálny model reliéfu Slovenska ASTER v 26x26m rozlíšení (prevzorkovaný na 50x50m) od spoločnosti ERSDAC, 2007,
- digitálny model reliéfu Slovenska EU DEM v 25x25m rozlíšení od spoločnosti EEA (European Environment Agency), 2013,
- digitálny model reliéfu okresov Nitra a Štúrovo v 10x10m rozlíšení, od spoločnosti ESPRIT spol. s r.o., Banská Štiavnica

- digitálny model reliéfu Tatranská Javorina v rozlíšení 0,2x0,2m vytvorený Mgr. Imrichom Jakobom, PhD., 2014.

Použité vektorové vrstvy:

- časti obcí Krásno nad Kysucou, Turzovka a Čadca – Bc. Dominika Kaisová (Kaisová, 2015),

- úseky riek Dunaj a Ipeľ pri normálovom a povodňovom stave – Mgr. Henrich Grežo, PhD. (Jakubcová, Grežo, Petrovič, 2014),

- vrstvy s vodnou nádržou a intravilánom obce Veľké Uherce – Bc. Patrik Belianský (Belianský, 2015),

- vrstvy s administratívnymi hranicami a riekou Nitra - ErasData - Pro s.r.o. Vzhľadom na to, že vrstva s riekou Nitra bola poskytnutá vo forme polygónu a do testu vstupovala línia, bolo potrebné vygenerovanie stredovej línie. Stredová línia bola vytvorená použitím Voronoiovho diagramu (*v.voronoi*).

Zvyšné vstupné dáta:

- databáza odchytených drobných zemných cicavcov - Mgr. Adam Konečný, Ph.D.,
- zoznam hradov Žilinského kraja – MUDr. Jozef Róza,
- ostatné vstupné dáta, ako zoznam tatranských vrchov či pomocné obalové zóny, sme vytvorili v priebehu tvorby skriptov podľa potreby jednotlivých testov.

2.5 Štruktúra opisu skriptov v práci

Opis skriptov v kapitole Výsledky a diskusia je rozdelený do šiestich logických častí pre každú kategóriu skriptov.

Prvá časť ma za úlohu informovať čitateľa o všeobecnej problematike danej kategórie. Pre ľahšie pochopenie načo jednotlivé skripty slúžia sú na začiatku každého opisu položené otázky, na ktoré dané skripty odpovedajú alebo na ktoré je možné s nimi odpovedať.

Druhá časť čitateľa informuje o prístupoch používaných pri riešení danej témy a stručná charakteristika fungovania skriptu.

Treťou časťou je opis vstupných dát. Tento opis je rozdelený na súborové vstupy a vstupy zadávané užívateľom. V tejto časti je čitateľ informovaný o podmienkach, forme a možnostiach vstupných dát. Charakteristika je doplnená autentickými obrázkami vstupov, zachytenými pri testovaní jednotlivých skriptov.

Ďalšia časť je venovaná samotnému algoritmu znázorneného pomocou vývojového diagramu spolu s popisom jednotlivých krokov.

V nasledujúcej časti je znázornený test algoritmu a jeho výsledky. Časť obsahuje parametre a vstupné dáta použité pri testovaní a popis jednotlivých výstupov. Opis je doplnený výstupmi v rôznych formách (grafy, obrázky, mapy, tabuľky, atď.).

Poslednou časťou je zhodnotenie a porovnanie výsledkov z testov spolu s nástrojmi podobného zamerania a zhrnutie zistených údajov spolu s možnosťou využitia v rôznych činnostiach.

Zdrojové kódy všetkých skriptov boli kvôli veľkosti presunuté do príloh.

3 VÝSLEDKY A DISKUSIA

Ako výsledok práce bolo vytvorených deväť bash shell skriptov rozdelených do štyroch hlavných oblastí podľa možností použitia:

1. Výpočet reálnej rozlohy
2. DBMS a tvorba priestorových údajov
3. Výpočet viditeľnosti pomocou paralelných výpočtov
4. Simulácia zaplavenia územia

3.1 Opis spoločných častí skriptov (VD. 1)

Keďže všetky skripty majú spoločnú štruktúru, sú v nich aj niektoré opakujúce sa časti (podmienky, funkcie, či nastavenia). Vzhľadom na to, že tieto časti tvoria základ pre správne fungovanie skriptov a sú obsiahnuté v každom z nich, bol pre ne vytvorený samostatný vývojový diagram s opisom jednotlivých logických celkov (VD. 1).

1. Dočasné súbory - prvým krokom je zmazanie dočasných súborov z predošlého spustenia a vytvorenie novej temp zložky (1).

2. Zenity - všetky skripty pre vytvorenie užívateľsky príjemnejšieho rozhrania využívajú nástroj Zenity. Keďže tento nástroj nie je súčasťou programových balíkov všetkých UNIX distribúcií, skript zisťuje, či sa tento nástroj v rámci spusteného systému nachádza (2). Ak nástroj chýba (3) a užívateľ potvrdí jeho inštaláciu, nasleduje overenie pripojenia (4), ak je počítač pripojený k sieti, tak sa Zenity doinštaluje (5) a algoritmus pokračuje. V opačnom prípade sa zobrazí chyba pripojenia a skript sa ukončí (Obr. 11a). Ak užívateľ inštaláciu nepotvrdí skript sa ukončí.

3. GRASS GIS - keďže skoro všetky skripty pre úplnú funkčnosť využívajú niektoré z GRASS GIS nástrojov, skript skontroluje či je spustený v GRASS terminále (6). Ak nie je, vyskočí chybové hlásenie a skript sa ukončí (Obr. 11b).

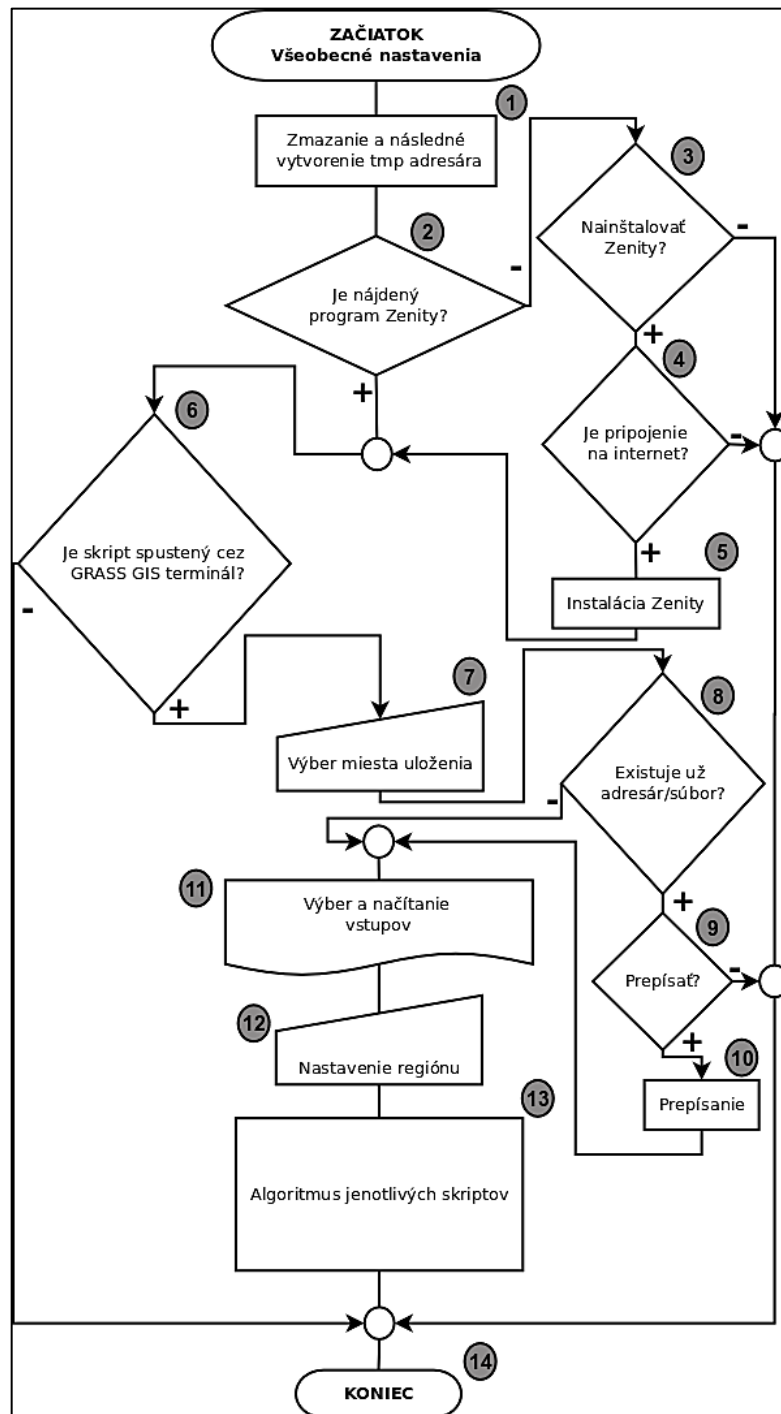
4. Výber miest uloženia - ďalším krokom je výber adresáru pre uloženie výstupov (7) (Obr. 11d). Ak sa v zvolenom mieste uloženia nachádza adresár alebo súbor s rovnakým názvom ako niektorý z výstupov daného skriptu (8), bude užívateľ upozornený na prepísanie týchto súborov (9). Potvrdením sa dané súbory prepíšu (10) (Obr. 11c).

5. Načítanie vstupných dát – na základe osobitných požiadaviek každého skriptu, prebehne výber a načítanie vstupných údajov (*r.in.gdal; v.in.org; a pod.*) (11) (Obr. 11e).

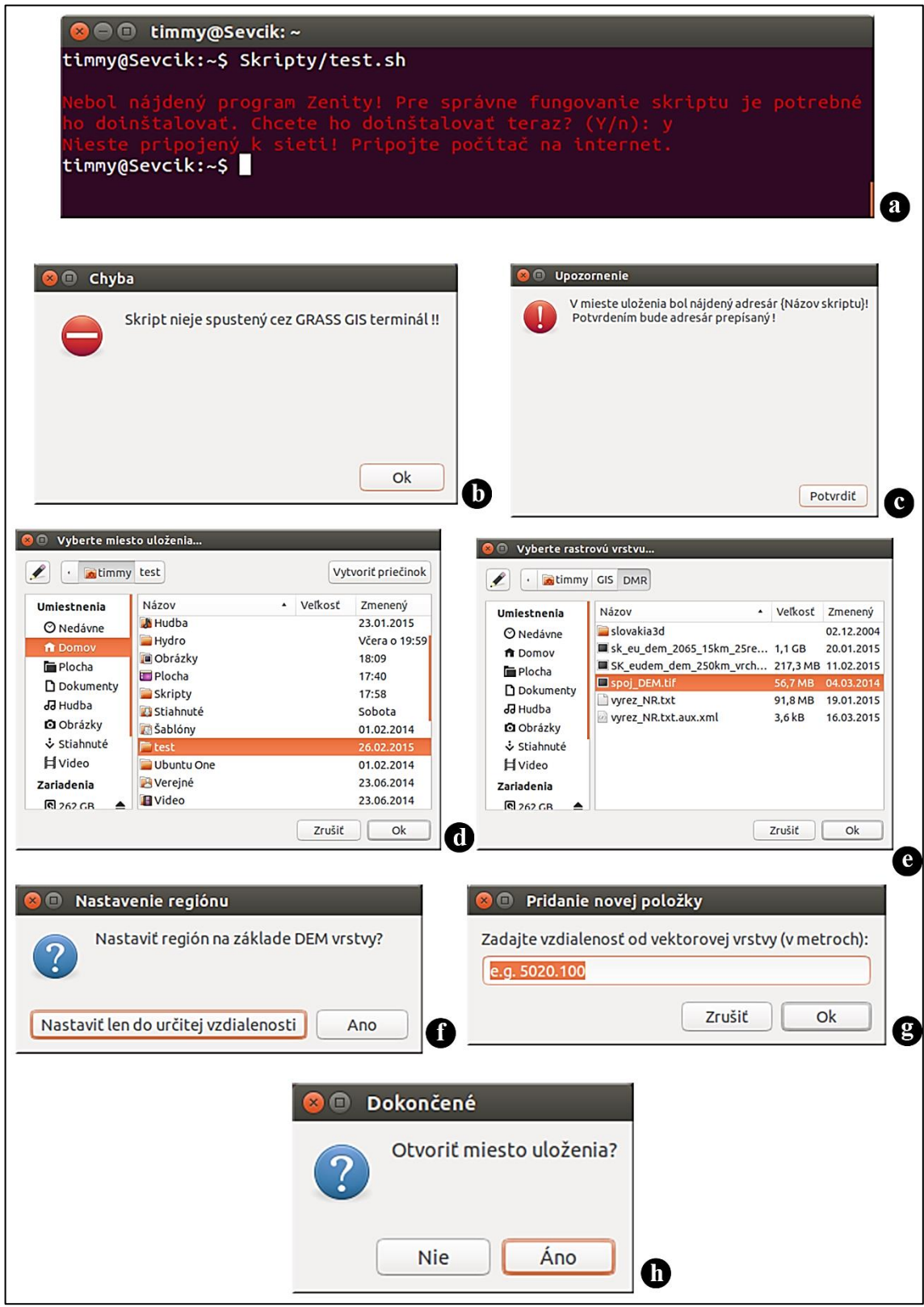
6. Nastavenie regiónu - následne prebehne nastavenie regiónu na základe vstupných vrstiev (*g.region*). Pri skripte Hydro.sh pribúda možnosť nastavenia regiónu len do určitej vzdialenosti od vektorovej vrstvy (12) (Obr. 11f,g).

7. Gro skriptu - od nastavenia regiónu pokračujú kroky algoritmov v opisoch jednotlivých skriptov (13).

8. Ukončenie - posledným krokom každého skriptu je informácia o ukončení s možnosťou otvoriť miesto uloženia (14) (Obr. 11h).



VD. 1 Vývojový diagram – Všeobecné nastavenia



Obr. 11 Všeobecné nastavenia

3.2 Výpočet reálnej rozlohy

Je možné vyrátať skutočnú 3-D rozlohu územia ?

Je klasický výpočet rozlohy dostačujúci ?

Aká je skutočná rozloha Slovenska ?

Rozloha patrí k jedným zo základných a najčastejšie využívaných geometrických údajov. Chápeme ju ako mieru určujúcu veľkosť ohraničenej plochy štandardne v štvorcových metroch.

Kolář, Pospíšilová (2007) definujú plochu v rámci euklidovského priestoru ako množinu všetkých bodov, ktorých súradnice vyhovujú rovnici: $F(x,y,z)=0$, kde F je funkcia, ktorá má v každom bode spojitú parciálnu deriváciu aspoň prvého rádu a na žiadnej otvorenej množine nie je identicky rovná nule.

Údaje o rozlohe sú súčasťou väčšiny dokumentov, koncepcií, plánov, stratégií či činností súvisiacich s tvorbou a ochranou životného prostredia, alebo akejkoľvek aktivity spojenej s využívaním krajiny. Rozloha sa v teoretickej ale aj praktickej rovine premieta do činnosti ako: pozemkové úpravy, skúmanie druhotnej krajinej štruktúry (DKŠ), štatistické výpočty (hustota populácie), plánovanie krajiny, evidencia pozemkov, výpočet ekologickej stopy atď.

1. Prístupy a opis stanovenia rozlohy

V súčasnosti umožňuje výpočet rozlohy, takmer každý GIS softvér:

- Quantum GIS - *Field calculator* - *\$area*
- Grass GIS – *r.surf.area*; *r.report / v.report option=area*
- AcrGIS - *Calculate Geometry – area*

Problémom je, že tieto nástroje sa k rozlohe daného územia chovajú ako ku 2-D objektom. Pri rastrových analýzach (v softvéroch ako napr. Grass GIS) je najčastejšie využívaným základným vzorcom na výpočet rozlohy: *počet stĺpcov * počet riadkov * plocha bunky bez rátania buniek s NULL hodnotou*. (GRASS Development Team, 2015).

Grass GIS pre výpočet rozlohy rastrov ponúka aj nástroj *r.surf.area*, ktorý počíta rozlohu pravidelných 3-D trojuholníkových bodov (generovaných v stredoch jednotlivých buniek).

Pre vektorové výpočty sa používa Gaussov vzorec plochy, čo je matematický algoritmus založený na samostatnom výpočte plochy jednotlivých častí polygónu a ich

následnom spočítaní (Dahlke, Rose, 2015). Sice existuje viacero vzorcov výpočtu rozlohy (Weingartenové vzorce plochy, Codazziho vzorce plochy atď.), princíp počítania rozlohy ako 2-D objektu zostáva nezmenený.

Hlavným rozdielom medzi výpočtom rozlohy pomocou vyššie uvedených nástrojov a našim skriptom je, že základným prvkom pre výpočet skutočnej rozlohy je reliéf, ktorý reálne odráža vzájomné pôsobenie endogénnych a exogénnych geomorfologických procesov. Základom pre takéto stanovenie rozlohy je potreba prekrytia skúmanej oblasti digitálnym modelom reliéfu (DEM). Na základe nadmorských výšok jednotlivých buniek vloženého DEM sa vypočíta sklon týchto buniek, čím získame približný obraz reliéfu daného územia. Takto pripravené vrstvy ďalej slúžia na výpočet skutočnej rozlohy.

Pre dosiahnutie čo najpresnejších výsledkov, bola do skriptu pridaná možnosť zmeny rozlíšenia. Toto nastavenie nijak neovplyvňuje kvalitu vloženého DEM (neprebíha prevzorkovanie rastra) ani samotný výpočet sklonitosti reliéfu. Umožňuje však výpočet rozlohy aj veľmi malých území, pri menšom rozlíšení DEM vrstvy s užívateľom zvolenou presnosťou (až na centimetre).

2. Vstupné dáta

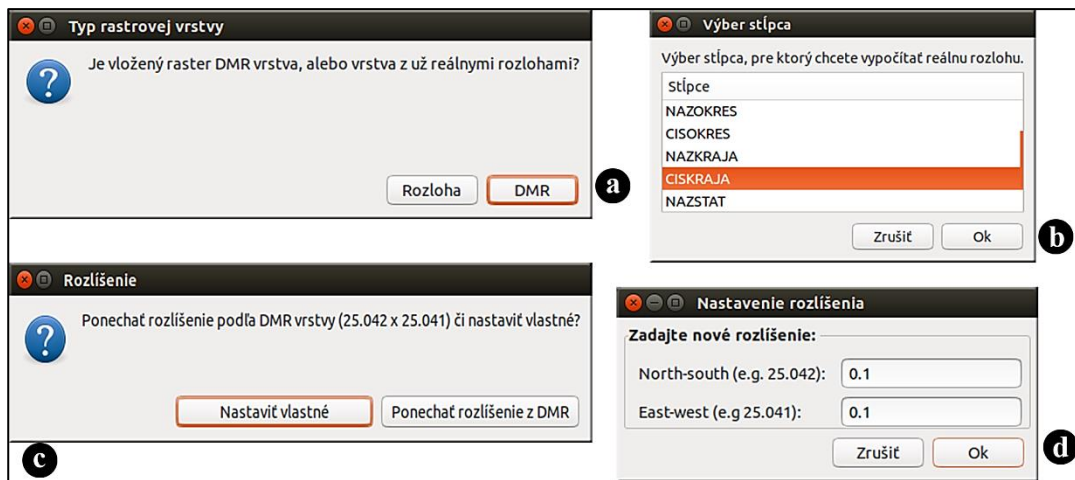
Do skriptu vstupujú dve vrstvy.

Prvou je polygónová vektorová vrstva záujmového územia, ktorého reálnu rozlohu chceme získať. Veľkosť tejto vrstvy nie je nijak obmedzená. Dané územie môže byť reprezentované ako jeden súvislý alebo viacero menších polygónov. Pre lepšiu prehľadnosť výsledkov v prípade viacerých polygónov je odporúčané mať v atribútovej tabuľke stĺpec, v ktorom sú polygóny z rovnakých kategórií označené rovnakými číselnými hodnotami (napr. polygóny reprezentujúce DKŠ: hodnoty 1-6). Ak tento stĺpec chýba, počíta sa s hodnotami stĺpca "cat". (Hodnoty stĺpca "cat", ako aj stĺpec samotný, vytvára GRASS GIS automaticky ako identifikátor jednotlivých geoprvkov pre každú vloženú vektorovú vrstvu.)

Druhou vstupujúcou vrstvou je rastrová vrstva, ktorá musí zodpovedať dvom podmienkam. Bunky musia obsahovať údaje o nadmorskej výške (DEM) a táto vrstva musí pokrývať celé územie zadané vektorovou vrstvou. Skript dokáže počítať s bunkami štvorcového aj obdĺžnikového tvaru.

Namiesto DEM je možné vložiť aj rastrovú vrstvu s už vypočítanými reálnymi rozlohami z predošlého počítania. Výsledkom pri tomto výbere bude textový dokument obsahujúci informáciu o rozlohe celej vlozenej vektorovej vrstve.

Zvyšné vstupy sú založené na báze interaktívnej komunikácie medzi skriptom a užívateľom prostredníctvom dialógových okien. Ako prvé si užívateľ zvolí typ vlozenej rastrovej vrstvy (Obr. 12a). Následne sa z atribútovej ponuky vektorovej vrstvy, vyberie stĺpec pre ktorý počítať rozlohu (Obr. 12b). Ďalším vstupom je výber rozlíšenia, podľa ktorého bude vektorová vrstva rasterizovaná a prebehne jej štatistický prekryv (Obr. 12c,d).



Obr. 12 Realna_Rozloha.sh - vstupy

3. Opis algoritmu (VD. 2)

Po všeobecnom nastavení skript ako prvé zisťuje, či je vložený raster DEM vrstva alebo vrstva s už vypočítanými reálnymi rozlohami (1). Ak vložená vrstva nie je DEM, tak prebehne rasterizácia (*v.to.rast*) a základná štatistika (*r.univar*) vloženého vektora s výpisom rozlohy do súboru "Rozloha.txt" (2). Ak vložený raster je DEM, proces pokračuje otázkou, či ponechať pôvodné rozlíšenie z DEM vrstvy alebo nastaviť vlastné (3). Podľa voľby skript užívateľa buď vyzve na zmenu rozlíšenia (4) alebo pokračuje výberom stĺpca, ktorý reprezentuje jednotlivé polygóny, a podľa ktorého bude rozloha počítaná (5).

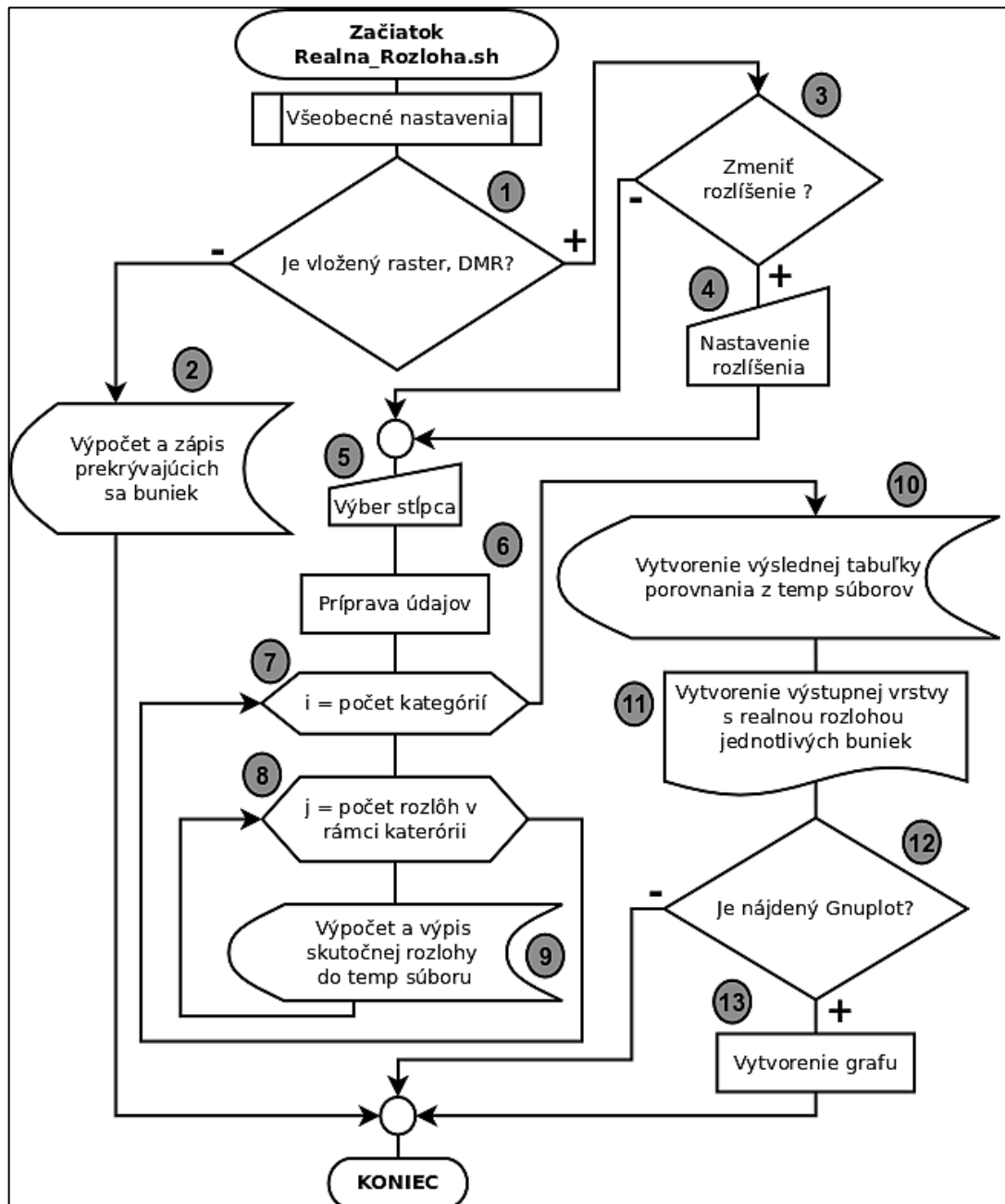
Ďalším krokom je príprava údajov (6). Tento krok zahŕňa prevod vektorovej vrstvy na rastrovú (*v.to.rast*), vytvorenie vrstvy sklonitosti (*r.slope.aspect*) a následný zápis štatistických prekryvov do dočasných súborov (*r.stats*).

Jadrom celého skriptu je samotný prepočet 2-D rozlohy na 3-D. Pre každú kategóriu podľa zvoleného stĺpca (7) prebieha výpočet a zápis skutočnej rozlohy do dočasných súborov (8) (9).

Po skončení cyklu nasleduje spracovanie dočasných súborov a vytvorenie výslednej tabuľky porovnania (10).

Ako ďalšie prebehne vytvorenie výstupnej vrstvy. Každá bunka vstupného rastra je reklasifikovaná (*r.recode*) na hodnotu rozlohy ktorú v skutočnosti predstavuje (11).

Ak je v spustenom systéme nájdený program Gnuplot (12) tak prebehne vytvorenie grafu porovnávajúceho pôvodné a výsledné reálne rozlohy (13).



VD. 2 Vývojový diagram – Realna_rozloha.sh

4. Test algoritmu a výstupy

Test bol zameraný na výpočet reálnej rozlohy Slovenska. Pre čo najpresnejšie vytýčenie hraníc bola použitá vrstva s administratívnymi hranicami Slovenska a DEM vrstva o rozlíšení 25 m pokrývajúca cele územie Slovenska.

Pri výbere typu rastrovej vrstvy bola vybraná "DEM" vrstva.

V prípade, ak vložená rastrová vrstva nesie informácie o skutočných rozlohách a pri výbere typu rastrovej vrstvy je označená "Rozloha", tak výsledkom bude ASCII súbor s rozlohou celej vlozenej vektorovej vrstvy

Pri výbere stĺpca, pre ktorý bude reálna rozloha rátaná, bol vybraný stĺpec s ID hodnotami krajov, v ktorom sú jednotlivým krajom priradené číselné hodnoty od 1 do 8. (1.- BA; 2.- TT; 3.- TN; 4.- NR; 5.- ZA; 6.- BB; 7.- PO; 8.- KE).

Pri nastavení rozlíšenia bolo ponechané pôvodné rozlíšenie vlozenej DEM vrstvy.

Prvým výstupom je tabuľka zobrazujúca pôvodné rozlohy (*r.stats*), novo vypočítané reálne rozlohy, ich rozdiel v metroch a v percentách pre každý kraj samostatne a pre všetky kraje spolu. (Tab. 2)

Tab. 2 Výsledná porovnávajúca tabuľka

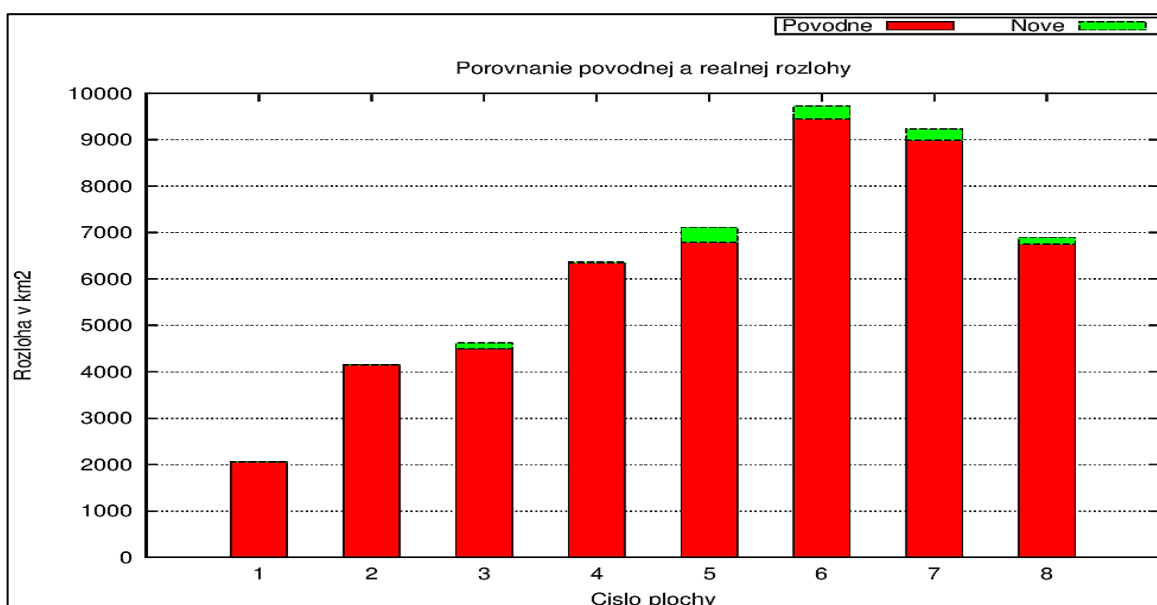
Názov_plochy	Pôvodná_rozloha(m2)	Reálna_rozloha(m2)	Rozdiel(m2)	Rozdiel v %
1	2053215725.012921	2065279337.013038	12063612.000117	.59
2	4145814536.533331	4158819373.199739	13004836.666408	.32
3	4501310198.196694	4622653323.708234	121343125.511540	2.63
4	6342029584.263597	6365894171.351536	23864587.087939	.38
5	6787561019.603806	7115228140.116552	327667120.512747	4.61
6	9452540275.477774	9727079784.781246	274539509.303473	2.83
7	8989538482.375393	9238669934.723364	249131452.347971	2.70
8	6750888255.975114	6891855152.395346	140966896.420232	2.05
Spolu	49022898077.438630	50185479217.289055	1162581139.850427	2.32

Druhým výstupom je rastrová mapa vloženého územia vo formáte GTIFF, v ktorej boli pôvodné hodnoty každej bunky reklasifikované hodnotou, ktorá zodpovedá reálnej rozlohe tej danej bunky v skutočnosti. (Obr. 13)

2400.8	2333.8	2285.5	2241.6	2198.5	2160.6	1474.5	1573.6	1298.6	1233.5	1175.4	1285.0
2367.0	2307.2	2262.4	2215.5	2177.4	2140.2	1326.9	1456.3	1312.7	1246.0	1154.0	1258.7
2376.3	2328.8	2282.9	2234.0	2194.8	2157.1	1123.6	1372.5	1404.8	1341.7	1221.2	1341.7
2378.0	2344.8	2301.3	2252.5	2211.8	2175.5	925.1	1175.4	1356.8	1341.7	1221.2	1356.8
Pôvodne hodnoty (DEM)						Reklasifikované hodnoty					

Obr. 13 Pôvodné a reklasifikované bunky

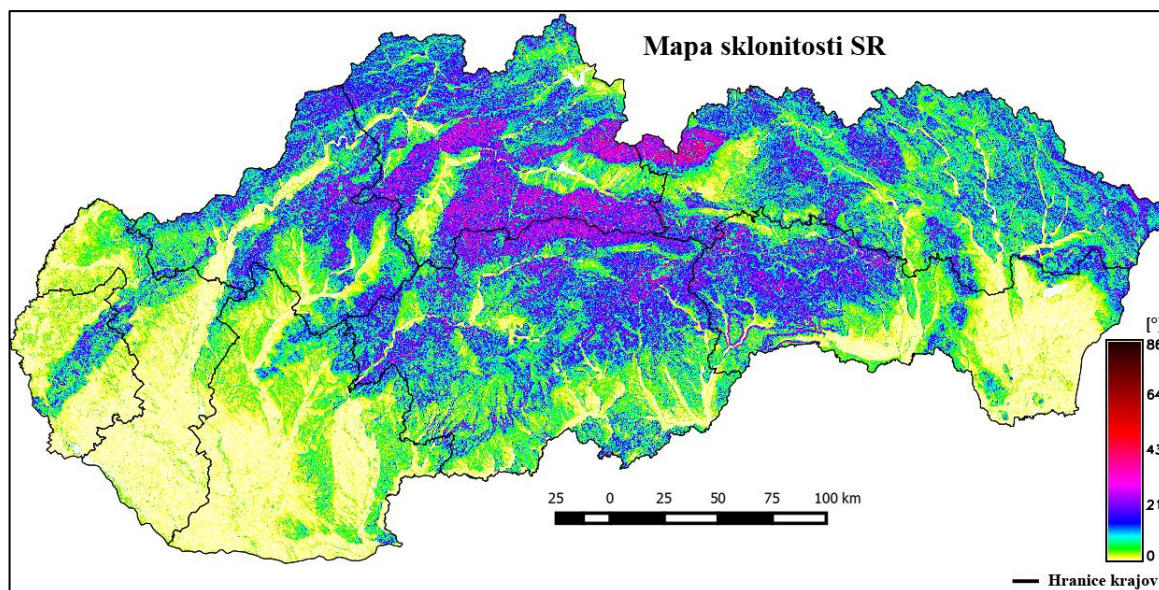
Ak v rámci systému, v ktorom je skript spustený, je nainštalovaný aj vykresľovací softvér Gnuplot, tak výsledkom bude aj graf porovnávajúci pôvodné a reálne rozlohy pre každú identifikačnú hodnotu vo zvolenom stĺpci (Obr. 14). Rozlohy vo výslednom grafe sú kvôli lepšej prehľadnosti premenené na km².



Obr. 14 Obrázok výsledného grafu porovnania, pôvodných a reálnych rozlôh

5. Zhodnotenie výsledkov a možnosti využitia

Z výsledného grafu (Obr. 14), či tabuľky (Tab. 2) je jasne vidieť, že vypočítaná reálna rozloha rastie na základe členitosti územia. Keďže v Žilinskom (5), Banskobystrickom (6) a Prešovskom (7) kraji sa nachádza väčšina slovenských pohorí, sú v nich zaznamenané aj najvyššie rozdiely. Rovnako to potvrdzuje aj obrázok 15 znázorňujúci rozdelenie sklonitosti SR.



Obr. 15 Sklonitosť SR

Pre porovnanie výsledkov vypočítanej reálnej rozlohy v závislosti od rozlíšenia bol test prevedený opätovne na záujmovom území Javorina s DEM vrstvou o rozlíšení 0,20 m.

Výsledné vypočítané reálne rozlohy z oboch testov sú porovnané s výsledkami iných GRASS GIS nástrojov na zisťovanie rozlôh v tabuľke 3.

Tab. 3 Porovnanie výpočtu rozlôh v 0,2m a 25m rozlíšení

	Typ metódy výpočtu	Javorina (0,20m) (m ²)	Slovensko (25m) (km ²)
<i>v.report</i>	2-D	3804.4406	49 022.9366
<i>r.report</i>	2-D	3804.4406	49 022.8980
<i>r.surf.area</i>	3-D	3957.6780	50 316.4300
<i>skript</i>	3-D	3903.4602	50 185.4792

* Nástroj *r.stats -a*, sa v tabuľke nenachádza z dôvodu, že na výpočet rozlohy, využíva rovnaký vzorec ako nástroj *r.report* a výsledky by boli identické.

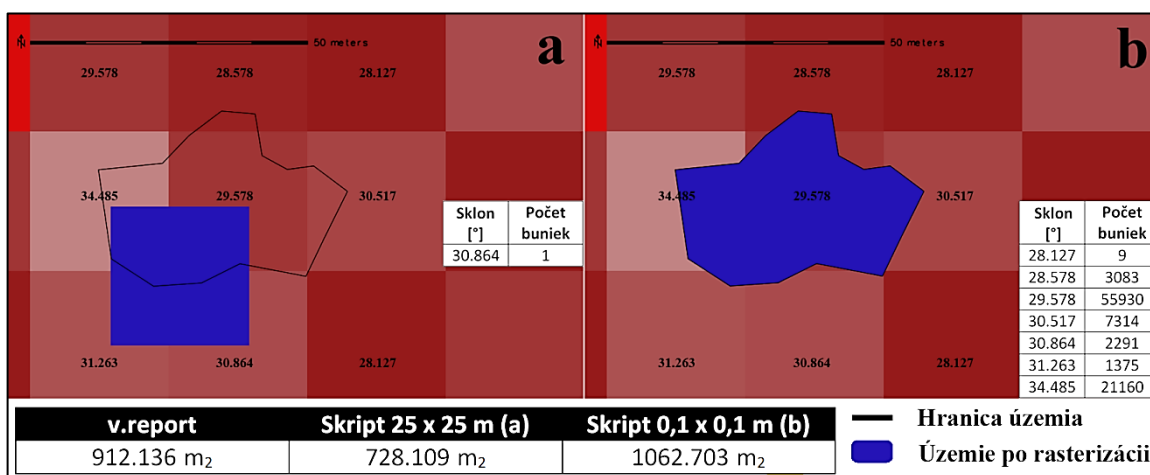
Keďže pri výpočte bolo ponechané pôvodné rozlíšenie DEM vrstvy, z porovnania nástrojov *v.report* a *r.report* je možné vidieť, že samotnou premenou vektorovej vrstvy na rastrovú vznikajú pri vyšších rozlíšeniach odchýlky spôsobené nedokonalým vyhladením okrajov vrstvy v rastrovej reprezentácii. Tieto odchýlky je možné odstrániť využitím možnosti nastavenia vlastného (vyššieho) rozlíšenia.

Za predpokladu, že hodnoty z nástroja *v.report* sú základom pre porovnávanie (100%), tak rozdiel medzi hodnotou vypočítanou skriptom a touto hodnotou je 99.0196 m² (2.54 %) pri Javorine a 1162.5426 km² (2.32 %) pri Slovensku.

Rozdiel hodnôt s nástrojom *r.surf.area* bol 153.2374 m² (4.03 %) pri Javorine a 1293.4934 km² (2,64 %) pri Slovensku. I keď výsledky nástroja *r.surf.area* sa k reálnej rozlohe blížia viac ako nástroje *v.report/r.report*, jeho najväčšími nevýhodami zostáva neschopnosť počítať rozlohu pre jednotlivé polygóny, čiže ráta len s rozlohou celej vrstvy, a chýbajúca možnosť zmeny rozlíšenia, kvôli čomu nedokáže počítať rozlohu menších území pri nižšom rozlíšení DEM.

Pre ukážku dôležitosti nastavenia rozlíšenia pri výpočte rozlohy menších území, pri nižšom rozlíšení bol prevedený test na vrstvách poskytnutých Bc. Dominikou Kaisovou, pri výpočte s pôvodným (25x25m) a zmeneným (0.1x0.1m) rozlíšením (Obr. 16).

Na obrázku 16a je vidieť chybu pri konvertovaní spôsobenú nesprávnym nastavením rozlíšenia a štatistiku prekryvu založenú na metóde dominantného typu (kapitola 2.3.2 - Rastre). Obrázok 16b znázorňuje výsledok konvertovania a štatistického prekryvu pri 0.1 m rozlíšení. Pre porovnanie výpočtu bola pridaná tabuľka rozlôh.



Obr. 16 Porovnanie 25m a 0.1m nastavenie rozlíšenia

Pre čo najväčšiu presnosť výsledkov:

- je vzhľadom k použitému analytickému softvéru (GRASS GIS) potrebné, aby vložená vektorová vrstva bola správne digitalizovaná, čiže neobsahovala chyby v geometrii. V opačnom prípade sa s niektorými polygónmi vôbec nemusí počítať,
- DEM vrstva by mala byť vyhladená, aby nedošlo k zbytočnému skresľovaniu spôsobeným depresiami alebo vyvýšeninami, ktoré v danom území v skutočnosti nie sú,

- faktorom ovplyvňujúcim čo najpresnejšie výsledky je aj rozlíšenie daného rastra. Čím vyššie rozlíšenie, tým bližšia podoba k reliéfu skutočného sveta. Avšak zvyšovaním rozlíšenia úmerne rastie aj doba výpočtov reálnej rozlohy,

- zmenou "precision=FCELL" na "precision=CELL" (100 riadok skriptu) pôjde výpočet niekoľkonásobne rýchlejšie, avšak za cenu mierneho skreslenia výsledku spôsobeného zaokrúhľením (float na integer).

Údaje o skutočných rozlohách je možno využiť v mnohých činnostiach a odvetviach. Od jednoduchého štatistického využitia na výpisy rozlôh katastrov, hustoty obyvateľstva, veľkosť využívania rôznych druhov pozemkom atď., až po riešenie zložitejších problémov, ako výpočet indexov (Index trvalo udržateľného ekonomického blahobytu, Index ľudského rozvoja), výpočet ekologickej stopy človeka, výpočet nákladov na odstraňovanie škôd po ekologických haváriách, atď.

Z hľadiska možnosti ďalšieho využitia je podstatná aj výsledná rastrová mapa, v ktorej jednotlivé bunky nesú informáciu o ich skutočnej rozlohe. V rámci mapy dáva užívateľovi možnosť zistiť reálnu rozlohu nových rozličných území, a to buď opätovným spustením skriptu s vložením vrstvy s rozlohou, alebo jednoduchým prekrytím záujmového územia a sčítaním hodnôt všetkých pokrývajúcich sa buniek.

3.3 DBMS a tvorba priestorových údajov

Dá sa urýchliť tvorba databázy ?

Je možné analyzovať niekoľko tisíc riadkovú databázu v špecifickej forme za pár sekúnd ?

Dá sa jednoducho a efektívne vytvoriť pravidelná bodová sieť s presne stanovenými parametrami ?

Databázy, ako základ pre modelovanie, tvorbu analýz, výstupov, ale aj pre vysoké finančné náklady na naplnenie údajmi, sú najkritickejšiu časťou nie len geograficko-informačných systémov (Hofierka, 2003).

Základnými jednotkami databáz sú údaje. GIS, ako odbor pracujú s osobitým typom údajov, a to s priestorovými údajmi (Huisman, Rolf, 2009). Existuje viacero rôznych definícií popisujúcich priestorové údaje, ale vo všeobecnosti môžeme povedať, že je to typ polohovo lokalizovaných údajov obsahujúcich tematické informácie spojené s údajom o polohe, pričom sú svojim geometrickým tvarom a polohou viazané na zemský povrch (Kusendová, 2003).

Prepojenie databázových údajov na polohovo lokalizované geometrické objekty má veľké využitie v mnohých oblastiach a činnostiach.

1. Prístupy a opis DBMS a tvorby sietí

Na prácu s databázami sa využívajú tzv. systémy riadenia databáz (database management system), čiže DBMS. Medzi DBMS zaraďujeme každý softvér umožňujúci jednoduchšiu tvorbu, ukladanie, zálohovanie, organizáciu, štatistické zapracovanie, či iné matematické a analytické operácie v rámci databáz (Maguire, 1991).

Existuje veľké množstvo rôznych typov databáz uložených v rozličných formátoch. Z toho dôvodu bolo vyvinutých viacero dopytovacích jazykov orientovaných na prácu so špecifickým typom databázy (napr. SQL, DMX, ISBL, OCL, atď.) (Narang, 2006).

Ak nie je použitá žiadna štandardná forma niektorého z dopytovacích jazykov, tak je možné povedať že každý užívateľ si môže zostaviť vlastnú formu databázy, pre ktorú je nutný individuálny prístup, a to najmä v prípade, ak je potrebné premietnuť databázové údaje do prostredia GIS a vytvárať priestorové údaje.

Náš skript pozostáva z troch samostatných a z jedného hlavného skriptu určeného na ich spustenie.

Prvý skript (Databaza.sh) využíva kombinatorickú analýzu na porovnanie každého riadku vlozenej databázy s každým, a následné dopočítanie časovej a priestorovej vzdialenosti z údajov porovnávaných riadkov. Tento skript bol vytvorený na základe žiadosti o prácu s už existujúcou databázou odchytených zemných cicavcov v špecifickej forme. Cieľom bolo porovnanie všetkých riadkov a zistenie časovej rozdielnosti uloženej v dvoch stĺpcoch vo formáte **09.08.** a **2011** a priestorovej vzdialenosti na základe kódov pascí vo formáte **H4** s 10 metrovými rozstupmi, kde písmená (A-H) tvoria stĺpce a číslice (1-8) riadky. Keďže žiadny z dopytovacích jazykov nedokáže vytvoriť požadovaný výsledok, na splnenie cieľa bol použitý Bash Shell skriptovací jazyk.

Druhý skript (Grid.sh) je zameraný na tvorbu pravidelnej bodovej siete predstavujúcej jednotlivé georeferencované body (pasce).

I keď GRASS GIS ponúka nástroj na vytváranie siete (*v.mkgrid*), jeho využitie je užitočne najmä pri tvorbe polygónových či líniových sietí. Nástroj *v.mkgrid* totiž pre vstup požaduje súradnice ľavého dolného rohu výstupnej mapy, pričom predpokladá, že od tohto bodu sa začína daná sieť v polygónovej či líniovej forme. Nedostatkom je, že v prípade bodovej siete sa postup nemení a výsledne body sú len pre-konvertované centroidy, čím vzniká posun celej siete o $n/2$ na osi x a y , kde n je vzdialenosť medzi dvoma bodmi. Keďže vstupné súradnice sú rohom celej mapy, zo zvyšujúcim sa naklonením siete sa tento posun postupne zvyšuje. Problémom je aj pomenovanie jednotlivých bodov vždy začínajúce od ľavého horného rohu nezávislé od požiadaviek užívateľa.

Síce nástroj *v.mkgrid* umožňuje vytvorenie bodovej siete, na požadovaný výsledný stav pre ďalšiu prácu je potrebné použitie viacerých nástrojov (napr. *v.edit*, *v.db.update*, atď.)

Naším cieľom bolo zefektívnenie tohto procesu a vytvoriť nástroj, ktorý na základe vstupných údajov od užívateľa zostaví pravidelnú bodovú sieť, ktorú následne premietne do GIS prostredia – Grid.sh. Pre čo najväčšiu presnosť a vyhnutie sa omylom bolo vytvorené tzv. “Základné zobrazenie siete“, ktoré sa užívateľovi zobrazí v terminále (Obr. 18j) a na základe ktorého skript pracuje a vytvára výslednú bodovú sieť.

Keďže skript pracuje v prostredí GRASS GIS, je potrebné zdôrazniť, že mapové jednotky (maps units) v spustenej projekcii musia byť metre. V opačnom prípade skript obsahuje podmienku, ktorá ho s chybovým hlásením okamžite ukončí (Obr. 18i).

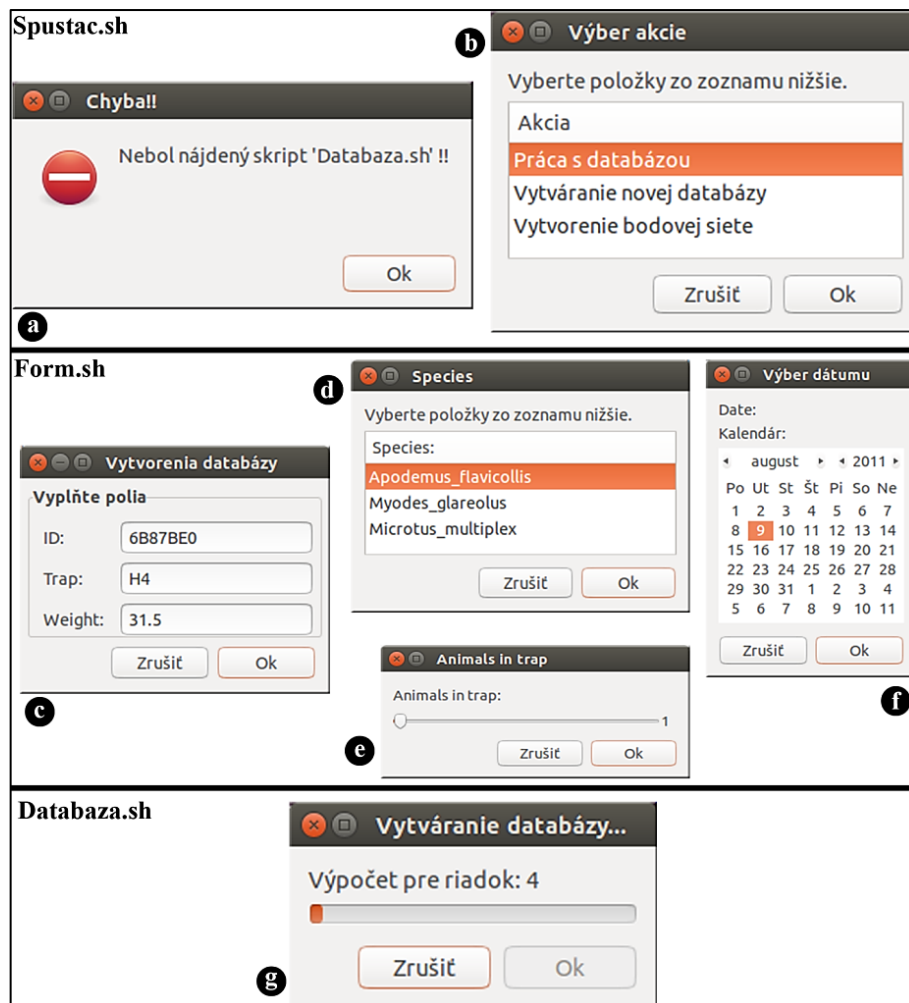
Tretí skript (Form.sh) je v podstate interaktívny formulár vytvorený na zjednodušenie tvorby, prípadne rozšírenie databázy vo forme vhodnej pre vstup do skriptu Databaza.sh.

2. Vstupné dáta

V rámci celého skriptu je jediným súborovým vstupom databáza do skriptu Databáza.sh, ktorá musí spĺňať niekoľko podmienok:

- stĺpce 2, 3 a 4 musia obsahovať kódy jednotlivých pascí a časový údaj v požadovanom formáte (vid. vyššie: *1. Prístupy a opis DBMS a tvorby siete*),
- v rámci poľa nesmú byť medzery (napr. Mus musculus = Mus_musculus),
- musia byť vyplnené všetky polia.

Vďaka zakomponovaniu nástroja sconvert (časť programového balíka Gnumeric), subskript podporuje desiatky rôznych formátov vstupnej databázy (.txt; .sxc; .ods; .xls; .xlsx; .dbf; .gnumeric...). Avšak ak v systéme, na ktorom je skript spustený, tento nástroj chýba, tak vložená databáza musí byť vo formáte ASCII text (.txt / .csv) s bodkočiarkou ";" ako oddeľovačom poľa.



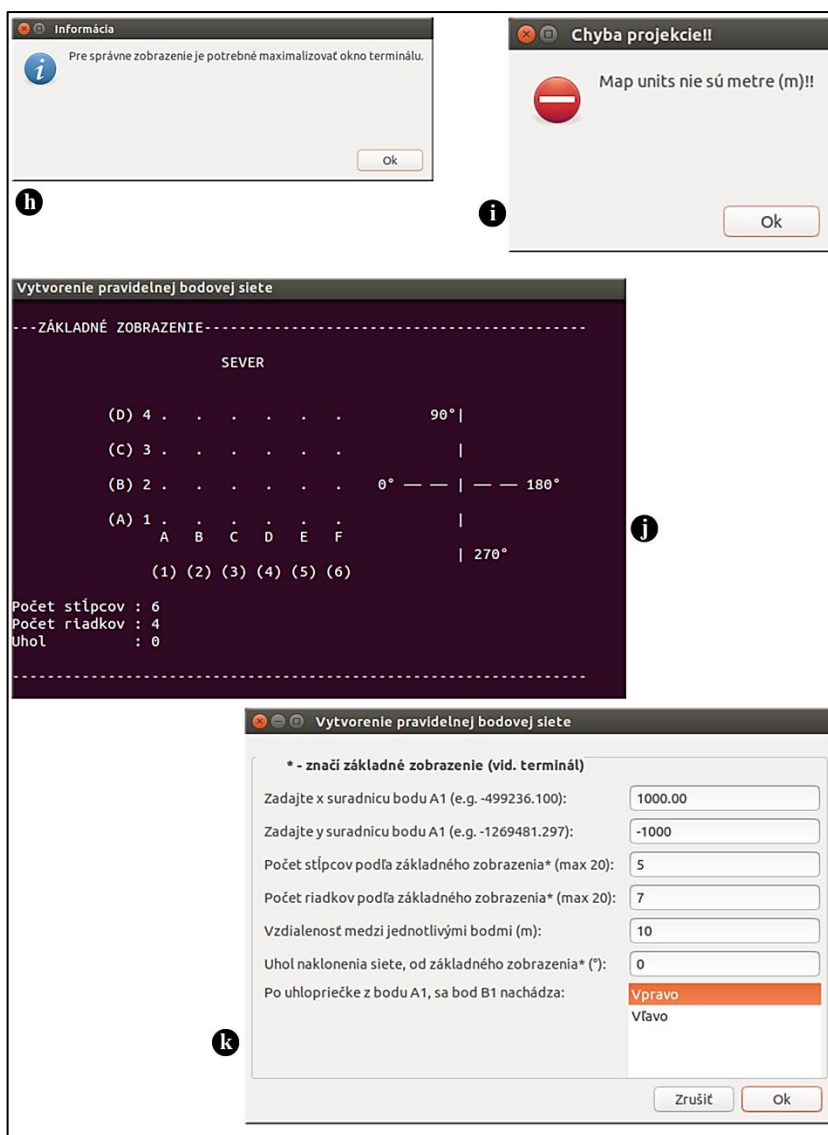
Obr. 17 Spustac.sh; Form.sh; Databaza.sh - vstupy

Zvyšné vstupy ostatných skriptov sú postavené na interaktívnej komunikácii s užívateľom prostredníctvom dialógových okien.

Pri skripte Spustac.sh je jediným vstupom výber akcie, podľa ktorej sa následne spustí daný skript (Obr. 17b).

Vstupom Form.sh skriptu je 15 samostatných interaktívnych okien, v ktorých užívateľ postupne vyberá údaje pre aktuálny riadok. Vyplňované údaje na základe vzorovej databázy: *ID; Trap; Date; Year; Session; Day of session; Altitude; Disturbance; Locality; Grid; Capture; Animals in trap; Rodent; Species; Weight; Sex; Mortality; Hookworm pres.* Na obrázku č. 17c-f sú zobrazené štyri vybrané vstupy.

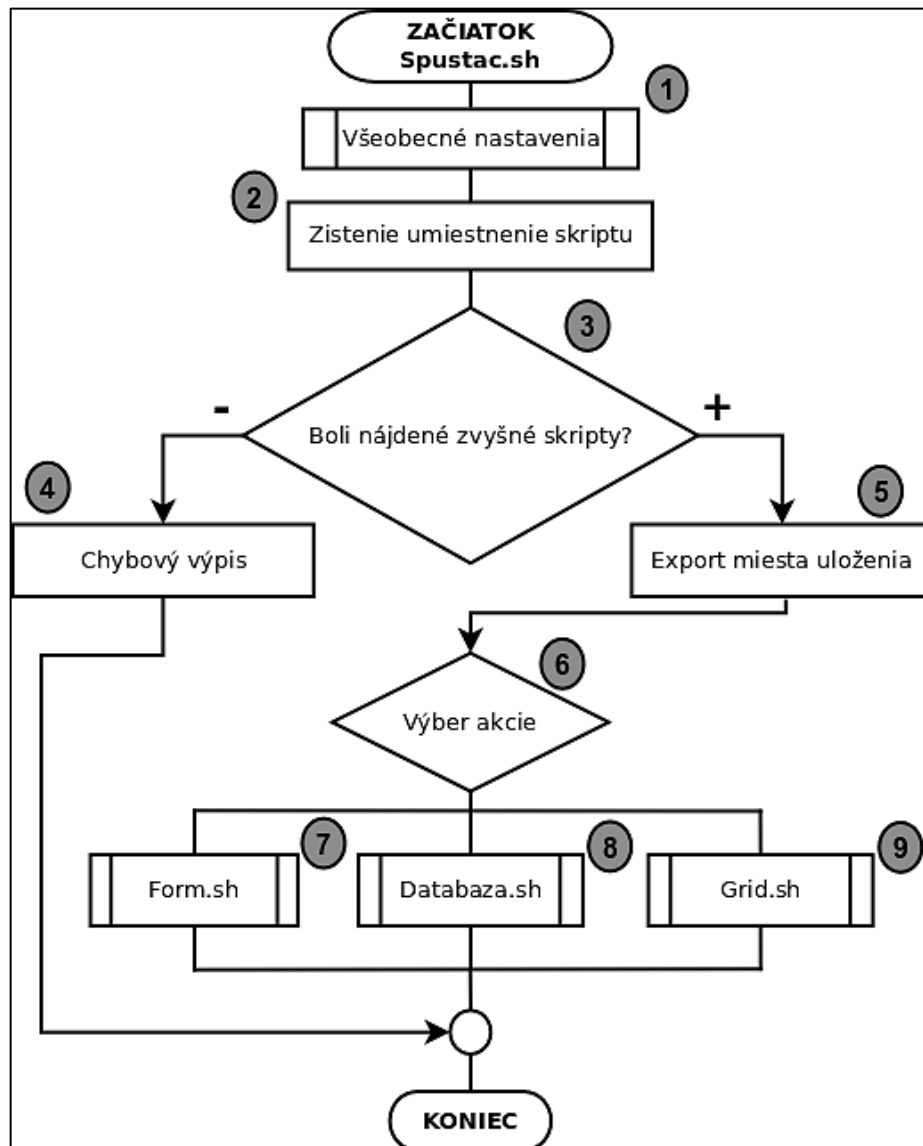
Jediným vstupom Grid.sh skriptu je tabuľka z obrázku č. 18k, kde si užívateľ zadáva súradnice počiatočného bodu (A1), počet stĺpcov a riadkov, vzdialenosť medzi bodmi, uhol naklonenia celej siete, a pozíciu bodu B1.



Obr. 18 Grid.sh - vstupy

3. Opis algoritmu

Spustac.sh - Vzhľadom na to, že výber skriptu, ktorý sa spustí závisí od aktuálnych potrieb užívateľa, bol vytvorený spúšťací skript “Spustac.sh“, ktorý zároveň pripravuje potrebné podmienky pre ostatné skripty (VD. 3).



VD. 3 Vývojový diagram – Spustac.sh

(1) Tento skript obsahuje len niektoré zo všeobecných nastavení. Zvyšné nastavenia sú rozdelené podľa osobitných nárokov do jednotlivých subskriptov.

Druhým krokom skript zisťuje cestu jeho umiestnenia (2).

Na základe zisteného umiestnenia prebehne kontrola, či sú všetky skripty v tom istom adresári (3). Ak niektorý chýba alebo sa nevolá pôvodným názvom zapísaným v popise každého skriptu, tak sa zobrazí chybová hláška a skript sa ukončí (4) (Obr. 17a). Ak boli nájdené všetky skripty, prebehne export premennej nesúcej údaj o mieste uloženia, pre

ostatné skripty (5) Ak sú skripty spustené samostatne a nie cez Spustac.sh, tak sa spustí výber miesta uloženia.

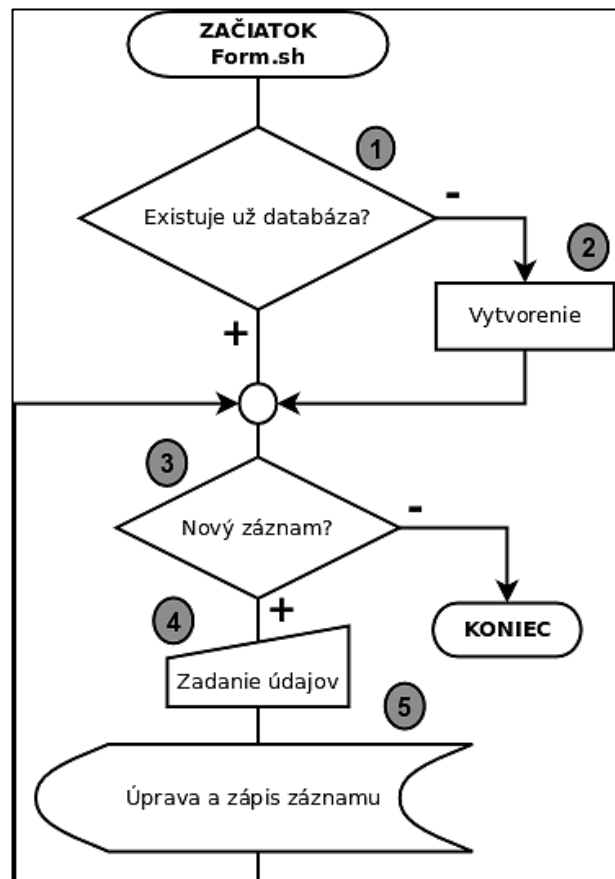
Ďalším krokom je samotný výber akcie, po ktorom sa otvorí príslušný skript (6) "Vytváranie novej databázy" – (7) "Práca s databázou" - (8) "Vytvorenie bodovej siete" - (9).

Form.sh – ide o interaktívny formulár na tvorbu databázy (VD. 4).

Ako prvé skript zisťuje, či vo zvolenom mieste uloženia existuje súbor "nova_databaza.txt" (1). Ak súbor existuje, tak pokračuje v jeho zápise, ak nie, tak vytvorí nový súbor s predvolenou hlavičkou (2).

Ďalším krokom je nekonečný cyklus (3), v ktorom užívateľ postupne zadáva údaje (4), ktoré sú následne upravované do výslednej formy a zapísané do výstupného súboru (5).

Cyklus sa preruší zrušením hlavného zadávacieho okna (Obr. 17c).



VD. 4 Vývojový diagram – Form.sh

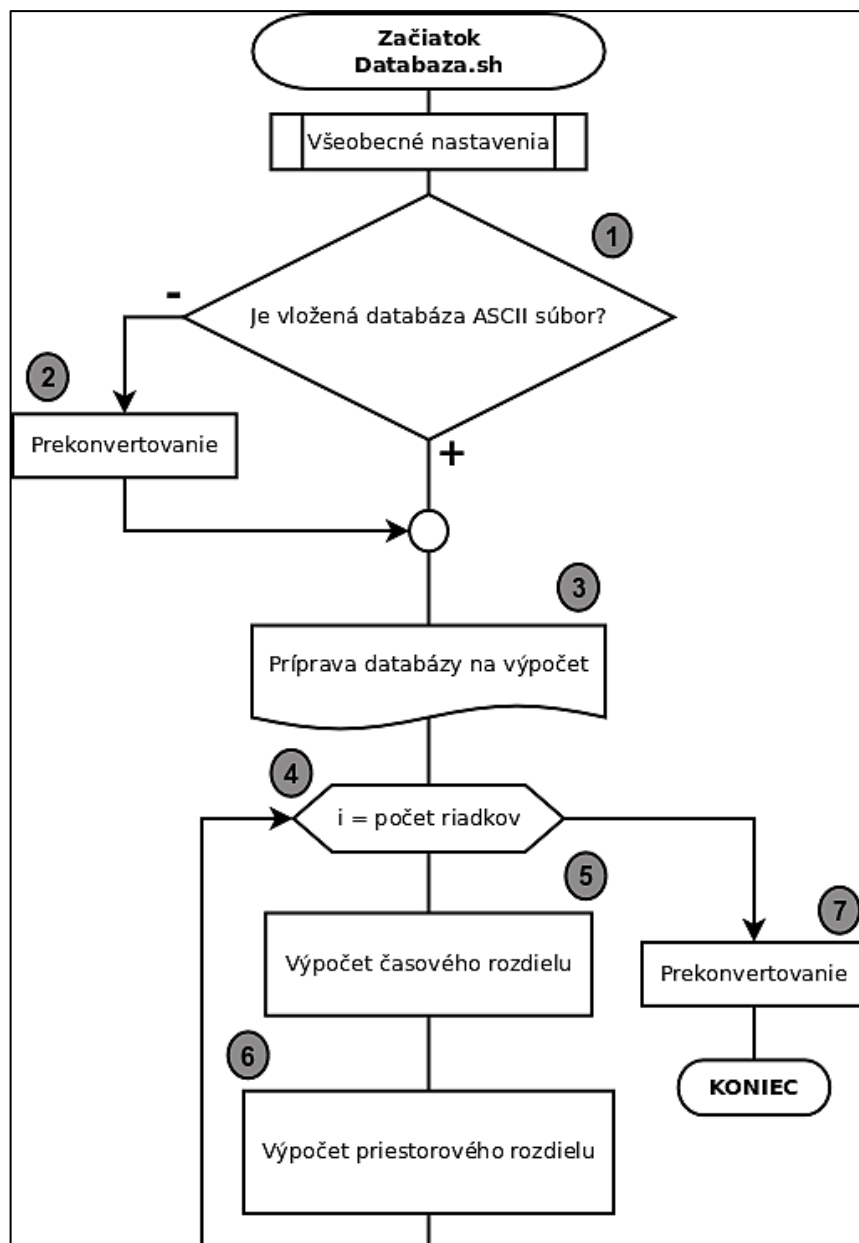
Databaza.sh – skript počíta priestorovú a časovú vzdialenosť porovnaní každého riadku s každým v už existujúcej databáze v špecifickej forme (VD. 5).

Ako prvé skript zisťuje súborový typ vlozenej databázy (1). Ak vložená databáza nie je v ASCII text forme (2), je do nej následne prekonvertovaná.

Ďalšou fázou je vytvorenie hlavičky a úprava databázy do potrebnej formy (3).

Po prípravách prebieha samotné porovnanie a výpočet časového (5) a následne priestorového (6) rozdielu pre každý riadok databázy osobitne (4) (Obr. 17g).

Posledným krokom je úprava databázy do výslednej podoby a jej prevod z ASCII text formátu do ODS formátu (7).



VD. 5 Vývojový diagram – Databaza.sh

Grid.sh – Skript vytvára presne georeferencovanú pravidelnú bodovú sieť (VD. 6).

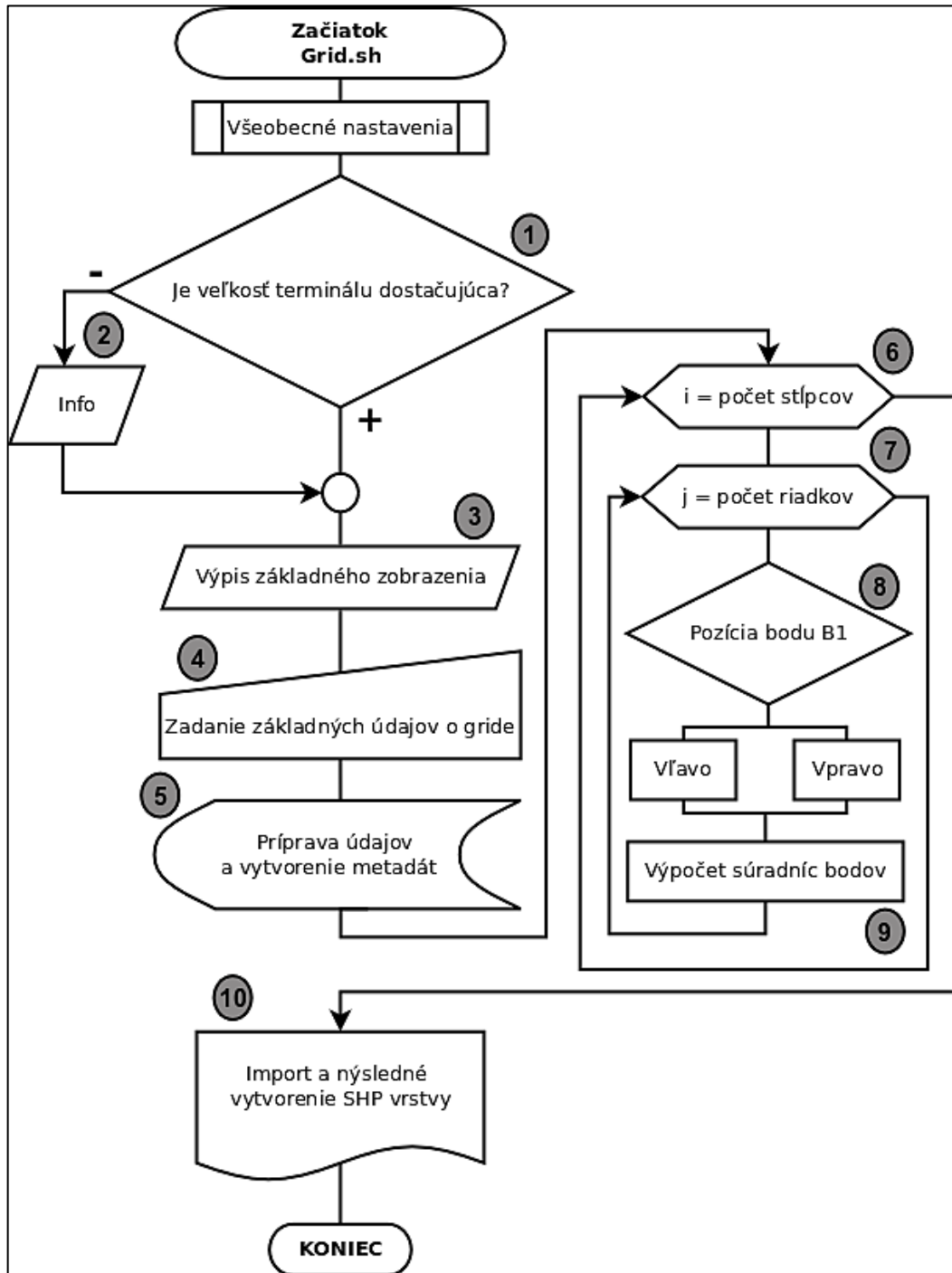
Keďže súčasťou skriptu je aj ukážka základného zobrazenia, podľa ktorého skript pracuje so vstupnými údajmi, je na začiatok skriptu vložená podmienka, ktorá kontroluje veľkosť okna terminálu (1). Ak je šírka terminálu nedostačujúca, zobrazí sa okno (2), ktoré užívateľa informuje o maximalizovaní terminálu (Obr. 18h). Následne prebehne výpis základného zobrazenia (3) (Obr. 18j).

V ďalšom kroku užívateľ vyplní formulár zo vstupnými údajmi (4).

Na základe zadaných hodnôt nasleduje úprava údajov do potrebnej formy a vytvorenie súboru s metadátami ku výslednému gridu (5).

S takto pripravenými údajmi skript pre počet stĺpcov (6) a riadkov (7) následne georeferencuje každý bod (9), a podľa zvolenej pozície bodu B1 ku každému priradí príslušné označenie (8).

Posledným krokom je načítanie vytvorených bodov do GRASS GIS (*v.in.ascii*) a následný export (*v.out.org*) do zvoleného miesta uloženia ako SHP vrstvu (10).



VD. 6 Vývojový diagram – Grid.sh

4. Test algoritmu a výstupy

Prvým testovaným bol skript Form.sh, ktorého výstupom je ASCII súbor "nova_databaza.txt" (Obr. 19a) vytvorený podľa vzorovej databázy (Obr. 19b). Táto novovzniknutá databáza ďalej slúžila ako vstup pre test skriptu Databaza.sh. Výsledkom testu je databáza vo formáte ODS (OpenDocument Spreadsheet) s požadovanými časovými (*day_difference*) a priestorovými (*spatial_difference_m*) rozdielmi medzi každým zápisom (Obr. 19c). V prípade absencie nástroja sconvert je výslednou databázou ASCII súbor "Databaza.txt" uložený v '/tmp/Databaza/'.

a

```

1 id;trap;date;year;session;day_of_session;altitude;disturbance;locality;grid;capture;animals_in_trap;rodent;SPECIES;weight;sex;mortality;hookwo
2 6B87BE0;H4;09.08.;2011;5;1;High;Disturbed;San_Giova;HD1;N;0;Mice;Apodemus_flavicolllis;"31,S";Male;no;0
3 6B87BE0;H5;10.08.;2011;5;2;High;Disturbed;San_Giova;HD1;W;1;Mice;Apodemus_flavicolllis;NA;Male;no;NA
4 6B87D8B;E5;18.05.;2011;2;1;High;Disturbed;San_Giova;HD1;N;1;Voles;Myodes_glareolus;"27,S";Female;no;0
5 6B88952;B3;13.07.;2011;4;1;High;Disturbed;San_Giova;HD1;N;1;Voles;Myodes_glareolus;"12,S";Male;no;1
6 6B88952;B5;14.07.;2011;4;2;High;Disturbed;San_Giova;HD1;W;1;Voles;Myodes_glareolus;NA;Male;no;NA
7 6B88F77;E5;19.05.;2011;2;2;High;Disturbed;San_Giova;HD1;N;1;Voles;Myodes_glareolus;"31,S";Male;no;0
8 6B89478;D1;19.05.;2011;2;2;High;Disturbed;San_Giova;HD1;N;1;Voles;Microtus_multplex;"24,S";Female;no;0
9 6B8979E;B2;14.06.;2011;3;1;High;Disturbed;San_Giova;HD1;N;1;Mice;Apodemus_flavicolllis;26;Male;no;0
10 6B8979E;C2;13.07.;2011;4;1;High;Disturbed;San_Giova;HD1;R;1;Mice;Apodemus_flavicolllis;32;Male;no;0
11 6B8979E;C6;14.07.;2011;4;2;High;Disturbed;San_Giova;HD1;W;1;Mice;Apodemus_flavicolllis;NA;Male;no;NA
    
```

b

	A	B	C	D	E	F	G	H	I	J	K	
1	id	trap	date	year	session	day_of_session	altitude	disturbance	locality	grid	capture	anim
2	6B87BE0	H4	09.08.	2011	5	1	High	Disturbed	San_Giova	HD1	N	
3	6B87BE0	H5	10.08.	2011	5	2	High	Disturbed	San_Giova	HD1	W	
4	6B87D8B	E5	18.05.	2011	2	1	High	Disturbed	San_Giova	HD1	N	
5	6B88952	B3	13.07.	2011	4	1	High	Disturbed	San_Giova	HD1	N	
6	6B88952	B5	14.07.	2011	4	2	High	Disturbed	San_Giova	HD1	W	
7	6B88F77	E5	19.05.	2011	2	2	High	Disturbed	San_Giova	HD1	N	
8	6B89478	D1	19.05.	2011	2	2	High	Disturbed	San_Giova	HD1	N	
9	6B8979E	B2	14.06.	2011	3	1	High	Disturbed	San_Giova	HD1	N	
10	6B8979E	C2	13.07.	2011	4	1	High	Disturbed	San_Giova	HD1	R	
11	6B8979E	C6	14.07.	2011	4	2	High	Disturbed	San_Giova	HD1	W	

c

	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	
1	capture	animals	rodent	SPECIES	weight	sex	mortality	hookwo	day_difference	spatial_difference (m)	id	trap	date	year	session	day_of	altitude	dis
2	N	1	Mice	Apodemus	31.5	Male	no	0	1	106887BE0H5	10.08.	2011	5	2	High	Dis		
3	N	1	Mice	Apodemus	31.5	Male	no	0	8331.62	6887D8B E5	18.05.	2011	2	1	High	Dis		
4	N	1	Mice	Apodemus	31.5	Male	no	0	2760.82	688952 B3	13.07.	2011	4	1	High	Dis		
5	N	1	Mice	Apodemus	31.5	Male	no	0	2660.82	688952 B5	14.07.	2011	4	2	High	Dis		
6	N	1	Mice	Apodemus	31.5	Male	no	0	8231.62	6888F77 E5	19.05.	2011	2	2	High	Dis		
7	N	1	Mice	Apodemus	31.5	Male	no	0	8250.00	6889478 D1	19.05.	2011	2	2	High	Dis		
8	N	1	Mice	Apodemus	31.5	Male	no	0	5663.24	688979E B2	14.06.	2011	3	1	High	Dis		
9	N	1	Mice	Apodemus	31.5	Male	no	0	2753.85	688979E C2	13.07.	2011	4	1	High	Dis		
10	N	1	Mice	Apodemus	31.5	Male	no	0	2653.85	688979E C6	14.07.	2011	4	2	High	Dis		
11	W	1	Mice	Apodemus	NA	Male	no	NA	84	306887D8B E5	18.05.	2011	2	1	High	Dis		
12	W	1	Mice	Apodemus	NA	Male	no	NA	2863.24	688952 B3	13.07.	2011	4	1	High	Dis		
13	W	1	Mice	Apodemus	NA	Male	no	NA	27	60688952 B5	14.07.	2011	4	2	High	Dis		
14	W	1	Mice	Apodemus	NA	Male	no	NA	83	306888F77 E5	19.05.	2011	2	2	High	Dis		
15	W	1	Mice	Apodemus	NA	Male	no	NA	8356.56	6889478 D1	19.05.	2011	2	2	High	Dis		
16	W	1	Mice	Apodemus	NA	Male	no	NA	5767.08	688979E B2	14.06.	2011	3	1	High	Dis		
17	W	1	Mice	Apodemus	NA	Male	no	NA	2858.30	688979E C2	13.07.	2011	4	1	High	Dis		
18	W	1	Mice	Apodemus	NA	Male	no	NA	2750.99	688979E C6	14.07.	2011	4	2	High	Dis		
19	N	1	Voles	Myodes	27.5	Female	no	0	5636.05	688952 B3	13.07.	2011	4	1	High	Dis		
20	N	1	Voles	Myodes	27.5	Female	no	0	57	30688952 B5	14.07.	2011	4	2	High	Dis		
21	N	1	Voles	Myodes	27.5	Female	no	0	1	06888F77 E5	19.05.	2011	2	2	High	Dis		
22	N	1	Voles	Myodes	27.5	Female	no	0	141.23	6889478 D1	19.05.	2011	2	2	High	Dis		
23	N	1	Voles	Myodes	27.5	Female	no	0	2742.42	688979E B2	14.06.	2011	3	1	High	Dis		
24	N	1	Voles	Myodes	27.5	Female	no	0	5636.05	688979E C2	13.07.	2011	4	1	High	Dis		
25	N	1	Voles	Myodes	27.5	Female	no	0	5722.36	688979E C6	14.07.	2011	4	2	High	Dis		
26	N	1	Voles	Myodes	12.5	Male	no	1	1	20688952 B5	14.07.	2011	4	2	High	Dis		
27	N	1	Voles	Myodes	12.5	Male	no	1	5536.05	6888F77 E5	19.05.	2011	2	2	High	Dis		
28	N	1	Voles	Myodes	12.5	Male	no	1	5528.28	6889478 D1	19.05.	2011	2	2	High	Dis		
29	N	1	Voles	Myodes	12.5	Male	no	1	29	10688979E B2	14.06.	2011	3	1	High	Dis		
30	N	1	Voles	Myodes	12.5	Male	no	1	014.14	688979E C2	13.07.	2011	4	1	High	Dis		
31	N	1	Voles	Myodes	12.5	Male	no	1	131.62	688979E C6	14.07.	2011	4	2	High	Dis		
32	W	1	Voles	Myodes	NA	Male	no	NA	56	306888F77 E5	19.05.	2011	2	2	High	Dis		
33	W	1	Voles	Myodes	NA	Male	no	NA	5644.72	6889478 D1	19.05.	2011	2	2	High	Dis		
34	W	1	Voles	Myodes	NA	Male	no	NA	30	30688979E B2	14.06.	2011	3	1	High	Dis		
35	W	1	Voles	Myodes	NA	Male	no	NA	131.62	688979E C2	13.07.	2011	4	1	High	Dis		
36	W	1	Voles	Myodes	NA	Male	no	NA	014.14	688979E C6	14.07.	2011	4	2	High	Dis		
37	N	1	Voles	Myodes	31.5	Male	no	0	041.23	6889478 D1	19.05.	2011	2	2	High	Dis		
38	N	1	Voles	Myodes	31.5	Male	no	0	2642.42	688979E B2	14.06.	2011	3	1	High	Dis		
39	N	1	Voles	Myodes	31.5	Male	no	0	5536.05	688979E C2	13.07.	2011	4	1	High	Dis		
40	N	1	Voles	Myodes	31.5	Male	no	0	5622.36	688979E C6	14.07.	2011	4	2	High	Dis		
41	N	1	Voles	Microtus	24.5	Female	no	0	2622.36	688979E B2	14.06.	2011	3	1	High	Dis		
42	N	1	Voles	Microtus	24.5	Female	no	0	5514.14	688979E C2	13.07.	2011	4	1	High	Dis		
43	N	1	Voles	Microtus	24.5	Female	no	0	5650.99	688979E C6	14.07.	2011	4	2	High	Dis		
44	N	1	Mice	Apodemus	26	Male	no	0	29	10688979E C2	13.07.	2011	4	1	High	Dis		
45	N	1	Mice	Apodemus	26	Male	no	0	3041.23	688979E C6	14.07.	2011	4	2	High	Dis		
46	R	1	Mice	Apodemus	32	Male	no	0	1	40688979E C6	14.07.	2011	4	2	High	Dis		
47																		

Obr. 19 Novovzniknutá (a), vzorová (b) a výsledná (c) porovnávacíaa databáza

Výstupom Grid.sh skriptu je SHP (shapefile) vrstva so samotnou bodovou sieťou a text súbor “Metadata.txt“ nesúci doplnkové informácie použité pri tvorbe vrstvy (Obr. 20).

```
Metadata.txt x
|
|-----Údaje zadané pri vytváraní vrstvy Grid.shp-----
Bod A1, súradnica x           : 1000.00
Bod A1, súradnica y           : -1000
Počet stĺpcov                  : 5
Počet riadkov                  : 7
Vzdialenosť medzi jednotlivými bodmi (m) : 10
Uhol naklonenia siete          : 0
Pozícia bodu B1                : Vpravo

Dátum vytvorenia vrstvy       : Po mar 23 23:00:15 CET 2015
Miesto uloženia vrstvy        : /home/timmy/Plocha/Grid
|-----
```

Obr. 20. Grid.sh – Metadáta k výslednej vrstve

5. Zhodnotenie výsledkov a možnosti využitia

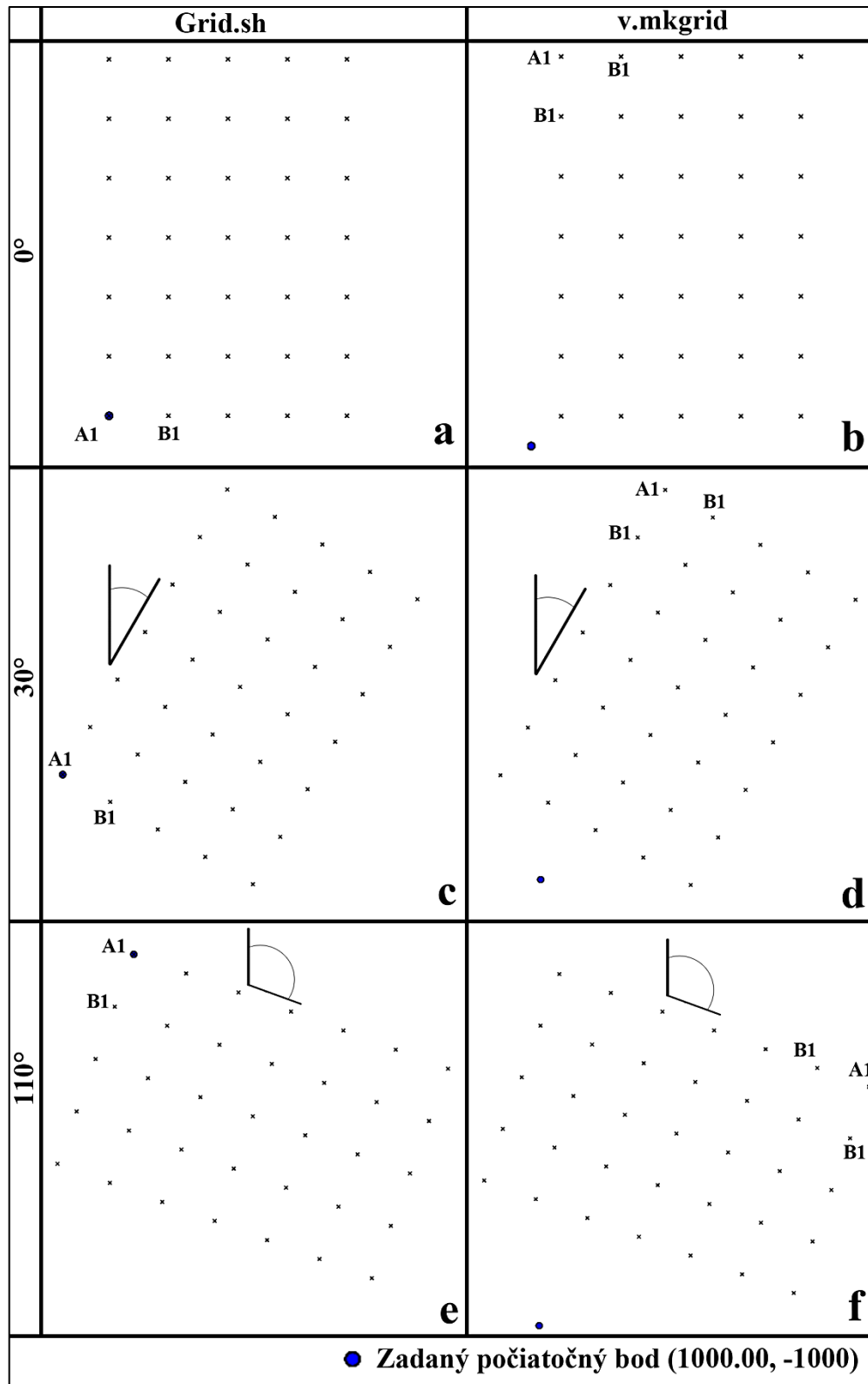
Databázy sú základnou časťou viacerých systémov. Na ich naplnenie potrebnými údajmi sa spotrebuje približne 50 – 80 % nákladov väčšiny projektov (Hofierka, 2003). Aj s toho dôvodu je pre čo najnižšie finančné či časové straty vhodné využiť možnosti automatizácie. Ručné postupné porovnávanie a výpočet pre každý riadok záznamu by bolo veľmi pracné, časovo náročné, náchylné k chybám a pri niekoľko tisíc riadkových databázach prakticky nemysliteľné.

Rovnako dôležité je aj zefektívnenie samotného procesu tvorby databázy. Skript Form.sh využíva interaktívne vkladanie údajov a ich následne uloženie v požadovanej forme, ktoré je pre bežného užívateľa z časového hľadiska násobne rýchlejšie.

Ako už bolo písané, každá databáza môže byť svojim obsahom a nárokmi užívateľa špecifická a vyžadovať tak individuálny prístup. I keď skripty Databaza.sh a Form.sh sú vytvorené pre potreby konkrétnej databázy s osobitými znakmi, poukazujú na možnosti využitia automatizácie pri náročnejších, ako aj každodenných bežných činnostiach.

Prevedenie údajov do GIS prostredia otvára nové možnosti od analýz až po vizualizáciu. Grid.sh skript vznikol ako nástroj na efektívnejšie vytvorenie pravidelnej bodovej siete pre jednoduchšie premietnutie priestorových údajov do prostredia GIS. Tvorba bodovej siete je súčasťou viacerých činností, ako napríklad odchyt živočíchov pomocou pascí, vytýčenie výskumnej plochy, či pri rôznych priestorových analýzach.

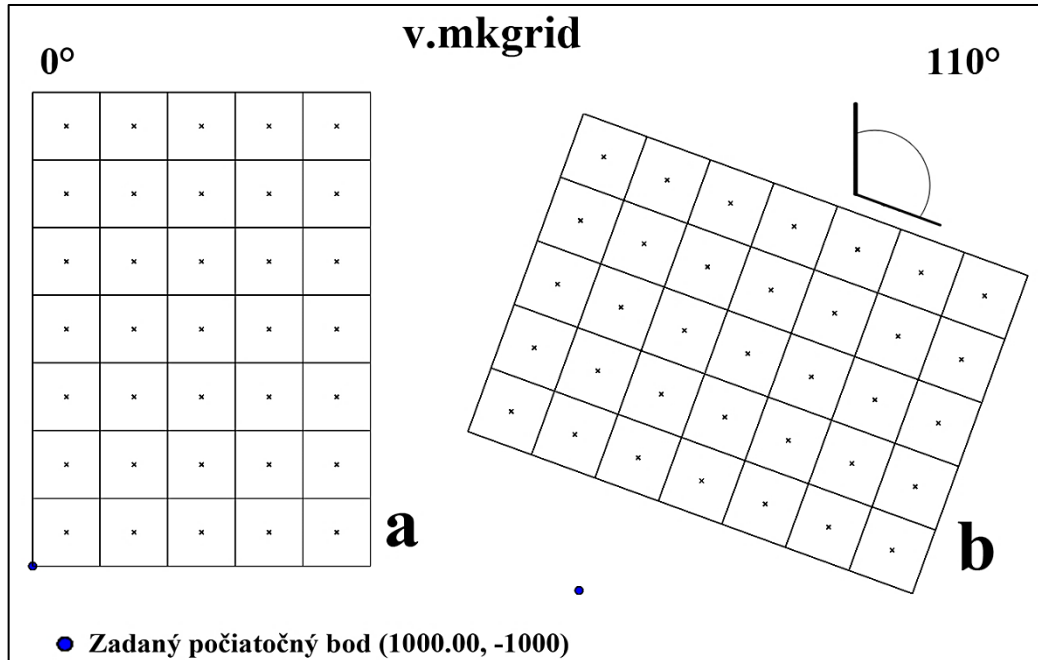
Pre porovnanie bol použitý aktuálny GRASS GIS nástroj na vytváranie siete - *v.mkgrid*. V oboch nástrojoch boli použité rovnaké vstupné údaje (poč. riadkov: 7; poč. stĺpcov: 5; súradnice: 1000,00,-1000; pozícia B1: *vpravo*), pri 0°, 30°, 110° (Obr. 21).



Obr. 21 Porovnanie siete z *v.mkgrid* a Grid.sh

Pri porovnávaní obrázkov sú vidieť rozdiely ako pri kódovaní jednotlivých bodov, tak aj pri umiestnení od zadaného počiatočného bodu.

Už spomínaný posun pri tvorbe bodovej siete v nástroji *v.mkgrid* je vidieť na obrázku č. 21b,d,f. Pre ilustrovanie dôvodu tohto posunu bol vytvorený obrázok č. 22.



Obr. 22 Posun siete pri *v.mkgrid*

Z obrázku č. 22a je vidieť, ako nástroj *v.mkgrid* vytvára sieť umiestnením bodov do geometrických stredov pomyselných polygónov. Obrázok č. 22b znázorňuje spôsob nakláňanie siete, a to otáčaním okolo vlastnej osi, čím dochádza k ešte väčšiemu posunu.

3.4 Výpočet viditeľnosti pomocou paralelných výpočtov

Je možné vypočítať viditeľnosť z akéhokoľvek miesta na Zemi ?

Akú časť územia je možné vidieť z najvyšších Slovenských vrchov ?

Dá sa zrýchliť proces výpočtu viditeľnosti ?

Koncept analýz viditeľnosti v krajine sa už dlhšiu dobu využíva vo viacerých oblastiach ľudskej činnosti, či už v armáde, archeológii, pri krajinnom plánovaní, hodnotení vizuálneho dopadu, atď. (Maloy, Dean, 2001; Miller, 2011). So stále rastúcim trendom využívania analýz viditeľnosti súvisí aj rozvoj technológií a zlepšovanie kvality priestorových údajov. Zo zvyšujúcou sa presnosťou získaných údajov rastie aj ich veľkosť, hardvérové nároky na systém, ale aj čas samotných analýz (Tabik, Zapata, Romero, 2012). S toho dôvodu bolo potrebné zavedenie určitých metód na zefektívnenie celého procesu, medzi ktoré patria aj paralelné výpočty.

1. Prístupy a opis výpočtu viditeľnosti

Výpočet viditeľnosti patrí ku základným priestorovým analýzám väčšiny GIS softvérov. Analýzy viditeľnosti sú založené na výpočte tzv. línie pohľadu (line-of-sight, a.k.a LOS). Viditeľnosť je definovaná ako LOS medzi dvoma bodmi (pozorovacím a cieľovým), pričom LOS nie je v žiadnom bode prerušovaná akoukoľvek časťou povrchu (Lee, 1991; Sorensen, Lanter, 1993 in Maloy, Dean, 2001). Výsledkom týchto analýz je väčšinou Boolerovský koncept viditeľnosti premietnutý do binárnej mapy (Fisher, 1996).

Hlavným limitujúcim faktorom využitia analýz viditeľnosti býva čas potrebný na ich výpočet. Kim, Rana a Wise, (2004), opisujú niekoľko možností zrýchlenia celého procesu:

- zníženie počtu vstupujúcich pozorovacích a cieľových bodov,
- nahradenie DEM vrstvy TIN vrstvou, ktorá na reprezentovanie reliéfu potrebuje menej bodov,
- ako pozorovateľov použiť len podmnožiny bodov pre každý cieľ.

Tak, ako vyššie spomenuté faktory, vplývajú na rýchlosť procesu aj použité algoritmy.

Z pohľadu DEM vrstiev sú najvyužívanejšími algoritmami (Ferreira et al., 2014):

a) **R3 algoritmus (Shapira, 1990)** - poskytuje priamy spôsob určenia viditeľnosti daného pozorovateľa v rámci záujmového polomeru. Z hľadiska času je však tento algoritmus značne pomalý a má využitie najmä pri menších územiach.

b) **RFVS algorithm (Franklin and Ray, 1994)** – ide o rýchlejší algoritmus počítajúci viditeľnosť buniek pozdĺž lúčov spájajúcich pozorovateľa (v strede bunky) so všetkými bunkami na hranici regiónu.

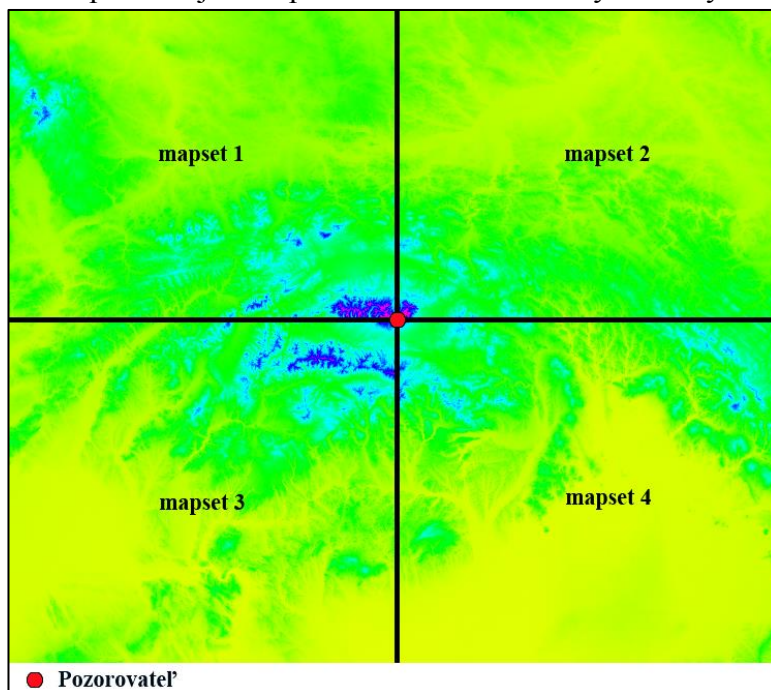
c) **Van Kreveld's algorithm (Van Kreveld, 1996)** – je ďalším rýchlym algoritmom s presnosťou ekvivalentnou R3 algoritmu. Na výpočet viditeľnosti využíva tzv. sweep line algoritmus, ktorý otáča líniu stredom zadaného bodu v protismere hodinových ručičiek, pričom zaznamenáva všetky bunky ktoré pretínajú túto líniu.

Síce existuje niekoľko autorov zaoberajúcich sa paralelnými výpočtami použitím R3 a RFVT algoritmov Osterman, (2012), Chao et al., (2011) a ďalší, je len málo výskumov zameraných na paralelne výpočty Van Kreveldovým algoritmom.

Naše skripty sú zamerané na zrýchlenie výpočtov na DEM vrstvách využitím paralelných výpočtov pomocou GRASS GIS nástroja *r.viewshed* založeného na Van Kreveldovom algoritme.

Náš skript pozostáva z dvoch na seba navzájom nadväzujúcich skriptov a z jedného hlavného skriptu určeného na spustenie analýz.

Cieľom prvého skriptu (*Shredder.sh*) je príprava a rozdelenie DEM vrstvy na štyri časti na základe vložených bodov reprezentujúcich pozorovateľa. Pre každý vložený bod prebehne vytvorenie štyroch mapsetov s rozličným nastavením regiónu podľa svetových strán a prekrývajúcich sa v šírke jednej bunky (Obr. 23). Skript umožňuje užívateľovi aj rozdelenie výsledných lokácií do n počtu adresárov pre prípad počítania na viacerých počítačoch.



Obr. 23 Hranice vytvorených mapsetov

Druhý skript (*Viditelnost.sh*) je zameraný na samotný výpočet viditeľnosti pomocou nástroja *r.viewshed*. V rámci tohto skriptu prebieha aj výsledné spojenie čiastkových máp z jednotlivých mapsetov a vytvorenie textovej časti výstupu.

Spúšťač skript (Parallel.sh) ma za úlohu postupné spustenie Viditelnost.sh skriptu pre každý mapset v samostatných termináloch. Využíva pritom vstavanú premennú prostredia GRASS_BATCH_JOB umožňujúcu spustiť GRASS GIS pomocou shell skript súboru.

2. Vstupné dáta

Vstupom pre Shredder.sh skript je DEM vrstva územia, podkladová polygónová vektorová vrstva a ASCII súbor s údajmi o pozorovateľovi.

DEM vrstva je základným faktorom ovplyvňujúcim presnosť výsledkov. Pre čo najdôkladnejšiu analýzu je potrebné, aby vstupná DEM vrstva bola riadne vyhladená – teda aby obsahovala čo najmenej výškových chýb.

Polygónová vektorová vrstva určuje hranice územia, ktoré bude rozdelené. Nevyžaduje sa žiaden špecifický stĺpec či atribút. Jedinou podmienkou je, aby bola vrstva bez chýb v geometrii.

ASCII súbor nesie informácie o jednotlivých pozorovacích bodoch. Je potrebné, aby spĺňal určité kritéria:

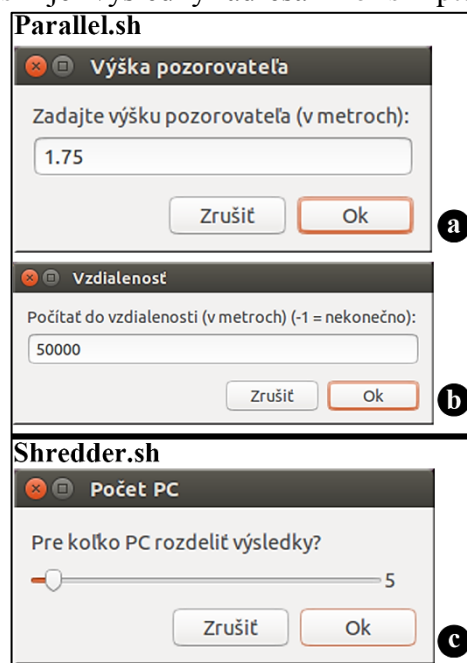
- musí obsahovať hlavičku,
- oddeľovačom poľa je čiarka,
- neobsahuje biele miesta (medzery, tabulátory),
- druhý a tretí stĺpec sú x a y súradnice daného pozorovateľa,
- všetky body sú obsiahnuté vo vlozenej polygónovej vrstve.

Jediným súborovým vstupom skriptu Parallel.sh je výsledný adresár zo skriptu Shredder.sh.

Zvyšné vstupy sú založené na báze interaktívnej komunikácie medzi skriptom a užívateľom prostredníctvom dialógových okien.

V skripte Parallel.sh sú jedinými takýmito vstupmi okná, v ktorých si užívateľ zadáva výšku, ktorá sa pripočíta k nadmorskej výške daného bodu – výška pozorovateľa (Obr. 24a) a vzdialenosť do ktorej bude výpočet prebiehať (Obr. 24b)

V skripte Shredder.sh užívateľ zadáva počet počítačov, na ktorých bude výpočet prebiehať (Obr.24c).



Obr. 24 Paralle.sh; Shredder.sh - Vstupy

3. Opis algoritmu

Shredder.sh – Skript pre každý vložený bod vytvára samostatnú GRASS lokáciu so vstupnou DEM vrstvou a štyrmi mapsetmi s presne nastavenými hranicami regiónov (VD. 7).

Prvým krokom skript od užívateľa zisťuje na koľkých počítačoch sa bude rátať (1). Tento krok je podstatný, ak sa ráta viditeľnosť viacerých bodov a výpočet nebude prebiehať na počítači s viac ako štyrmi jadrami, respektíve na multi-core servere. Ak však výpočet bude prebiehať na viac jadrových počítačoch či serveroch, užívateľ si môže navoliť počet pozorovacích bodov na jeden PC a tým aj počet využitých CPU.

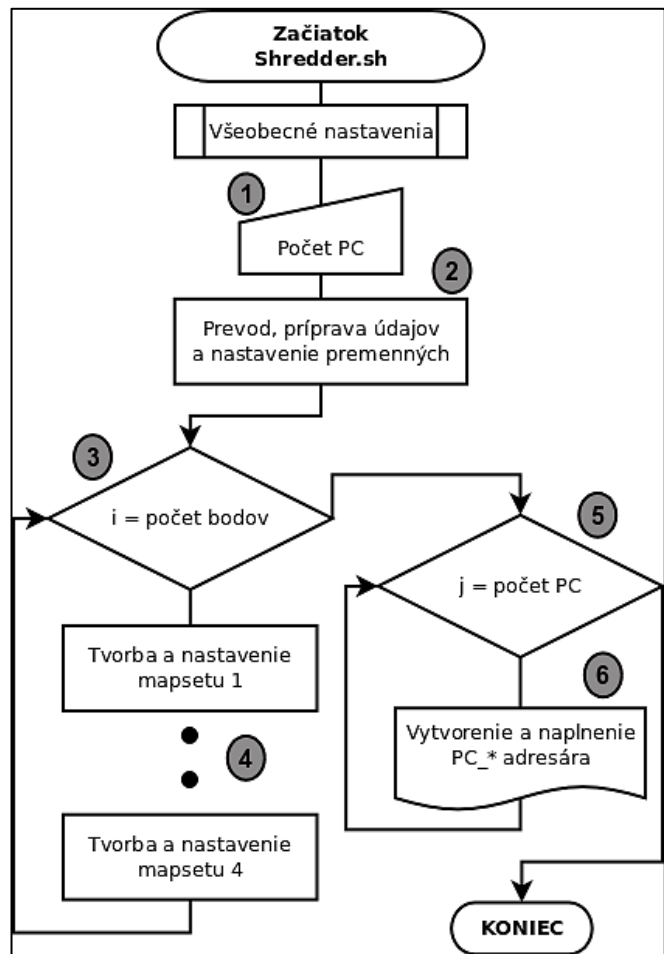
Druhým krokom je prevod (*v.to.rast*) a príprava vložených vrstiev do potrebnej formy a následné nastavenie premenných so základnými hranicami územia (2).

Po príprave vrstiev a premenných nasleduje samotné rozdelenie hraníc (*g.region*) do vytvorených mapsetov (*g.mapset*), pre každý vložený pozorovací bod (3).

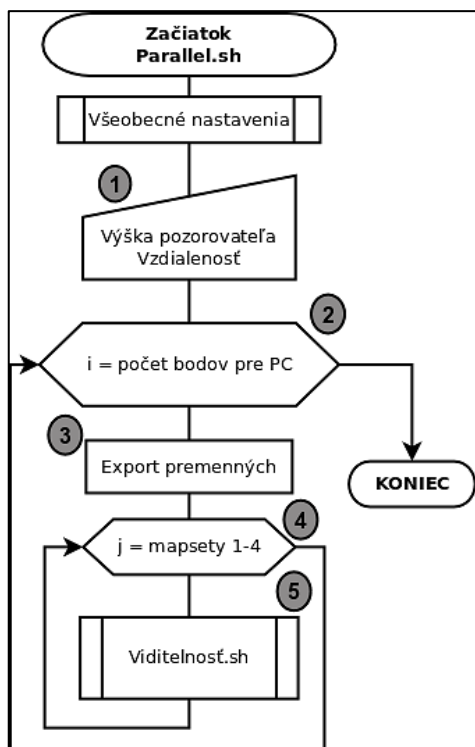
Novovzniknuté hranice sú pre každý mapset tvorené pôvodnými hranicami vektorovej vrstvy a súradnicami daného pozorovacieho bodu zväčšenými o polovičnú dĺžku rozlíšenia vlozenej DEM vrstvy (4).

Výsledné lokácie každého bodu sú rozdelené do výsledných adresárov zvoleného počtu PC a ku každému je vytvorený ASCII súbor informujúci aké body sa v danej PC zložke nachádzajú (5). Ak pri rozdelení bodov do PC adresárov nejde o bezzvyškové delenie, tak zvyšné lokácie sú vzostupne zatriedňované do jednotlivých adresárov (6).

Pre následné počítanie viditeľnosti je potrebné do výslednej PC zložky vložiť skript *Viditelnost.sh*



VD. 7 Vývojový diagram – Shredder.sh



VD. 8 Vývojový diagram – Parallel.sh

Viditelnosť.sh – Úlohou skriptu je vypočítať viditeľnosť pre spustenú oblasť a následne tieto oblasti spojiť (VD. 9).

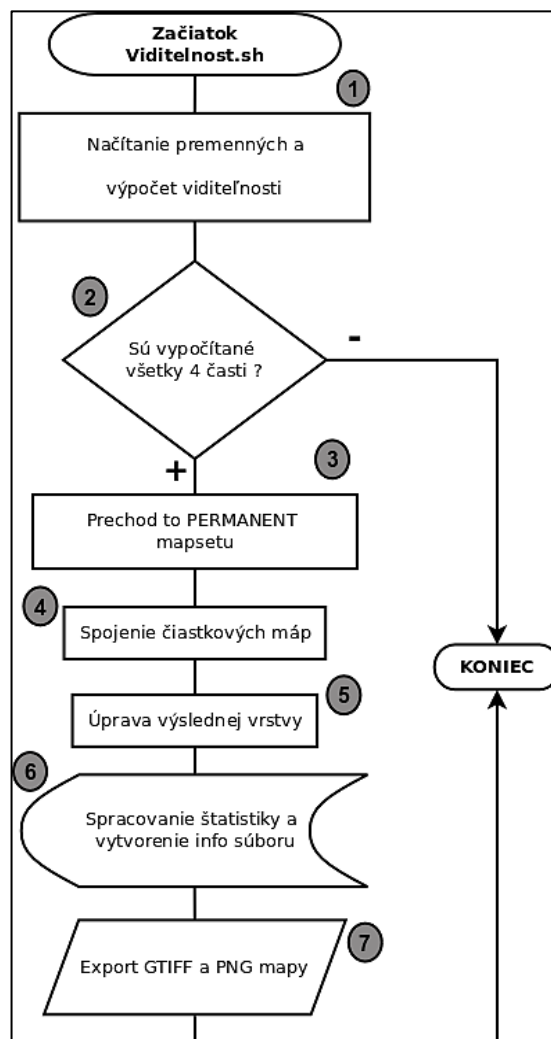
Po načítaní premenných prebehne samotný výpočet viditeľnosti (*r.viewshed*) pre aktuálny kvadrant daného bodu (1).

Po skončení rátania je zisťované či bol aktuálny vypočítaný kvadrant posledným (2). Ak áno, skript zmení pracovné prostredie na PERMANENT mapset (3) a pomocou funkcie *r.patch* spojí čiastkové mapy zo zvyšných mapsetov do mapy celého územia (4). Spojená vrstva viditeľnosti celého územia je následne upravená do výslednej podoby (*r.mapcalc*) (*r.colors*) (5). V ďalšom kroku z výslednej mapy prebehne štatistika prekryvu (*r.stats*) a všetky vstupné a zistené výstupné údaje sa zapíšu do info.txt súboru (6).

Parallel.sh – Jeho hlavnou úlohou je paralelné spustenie skriptu Viditelnosť.sh pre každý mapset, všetkých bodov (VD. 8).

Po vložení výstupného adresára zo skriptu Shredder.sh skript zisťuje, akú výšku pozorovateľa použiť a do akej vzdialenosti viditeľnosť počítať (1). Ak sa pri voľbe vzdialenosti zapíše hodnota “-1” tak je viditeľnosť rátaná pre celý región.

Po zadaní týchto základných údajov sa spustí gro celého skriptu a to, cyklus (2), ktorý pre každý bod vo vloženej PC zložke exportne potrebné premenné (3) a rozbehne ďalší cyklus, ktorý pre mapsety 1 až 4 (4) spustí skript Viditelnosť.sh (*GRASS_BATCH_JOB*) (5).



VD. 9 Vývojový diagram –Viditelnosť.sh

Posledným krokom je vytvorenie GTIFF súboru (*r.out.gdal*) viditeľnej časti územia a PNG mapy viditeľného územia spolu s DEM podkladom (*d.mon*) (7).

4. Test algoritmu a výstupy

Na testovanie skriptov bola použitá DEM vrstva (EEA) s rozlíšením 25x25m a ako pozorovací bod bol vybraný Gerlachovský štít.

Vstupné parametre:	Výška pozorovateľa	1.75 m
	Vzdialenosť	50 000 km
	Počet PC	1

Prvým testovaným bol skript Shredder.sh. Výstupom tohto skriptu je *n* počet GRASS lokácií, pričom každá lokácia obsahuje 4 mapsety s rozličným nastavením hraníc regiónu (Obr. 23) a PERMANENT mapset so vstupnou DEM vrstvou. Druhým výstupom je ASCII súbor s informáciou, o aké pozorovacie body v rámci danej PC zložky ide.

Výsledná zložka zo skriptu Shredder.sh slúži ako vstup pre skripty Parallel.sh a Viditelnost.sh. Ich výstupom je samostatná zložka pre každý pozorovací bod obsahujúca tri súbory. Prvým je textový súbor so základnými a získanými údajmi (Obr. 25).

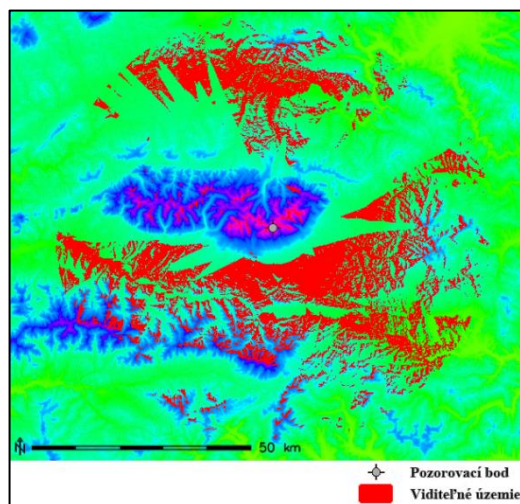
Gerlach-info.txt ×
Gerlach
Súradnice: -342341.321528, -1185937.40753
Použitá výška: 1.75 m
Do vzdialenosti: 50000 m
Rozloha celého regiónu: 7850.000 km ² (100 %)
Rozloha viditeľnej časti: 1810.202 km ² (23.05 %)
Sú vidieť tieto pozorovacie body:

Informácia o ostatných viditeľných bodoch vychádza zo vstupného ASCII súboru. Keďže pre test bol vybraný len jeden pozorovací bod, nie sú vidieť žiadne iné body.

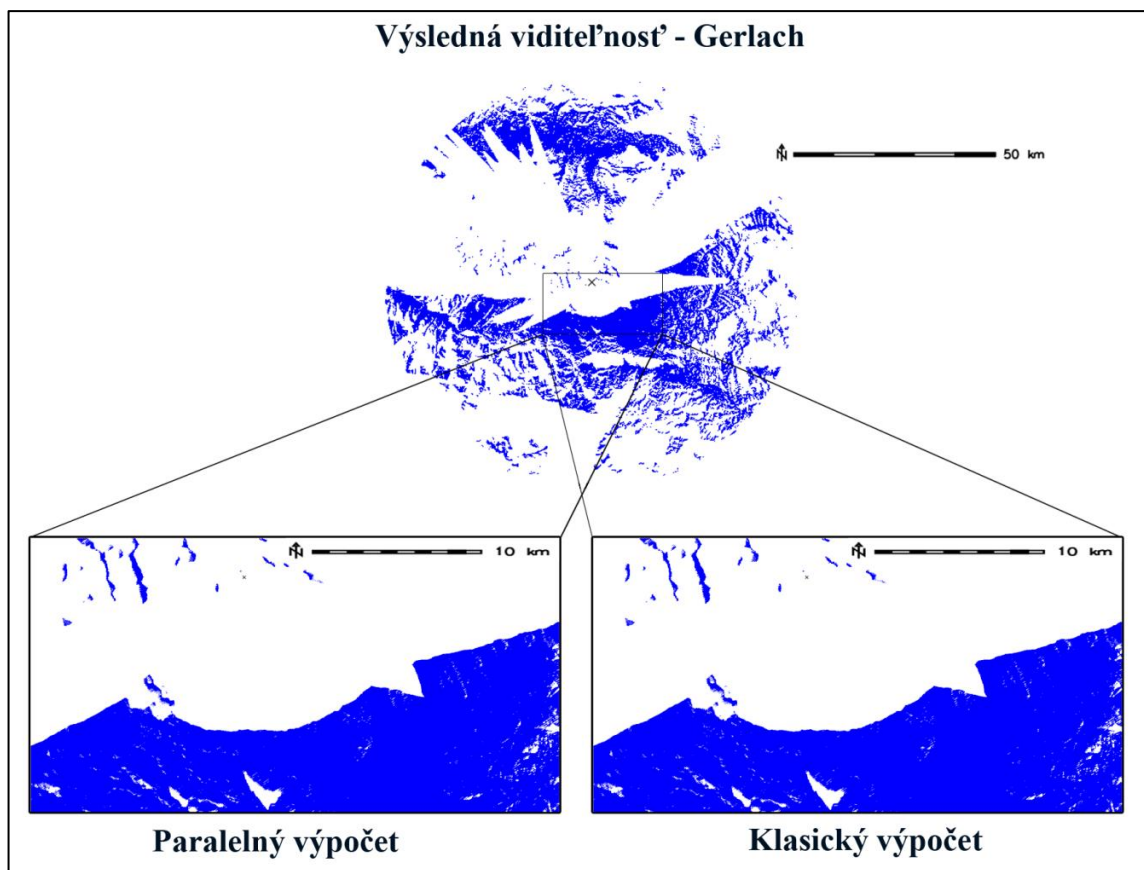
Obr. 25 Viditelnost.sh – textový výstup

Druhým súborom je exportovaná mapa viditeľnosti vo formáte PNG s rozlíšením 640x480 pixlov (Obr. 26). Táto mapa slúži len ako náhľad a nie ako výsledná mapa viditeľnosti.

Tretím súborom je výsledná rastrová vrstva vo formáte GTIFF obsahujúca výslednú viditeľnosť pre daný bod (Obr. 27).



Obr. 26 Viditelnost.sh – PNG výstup



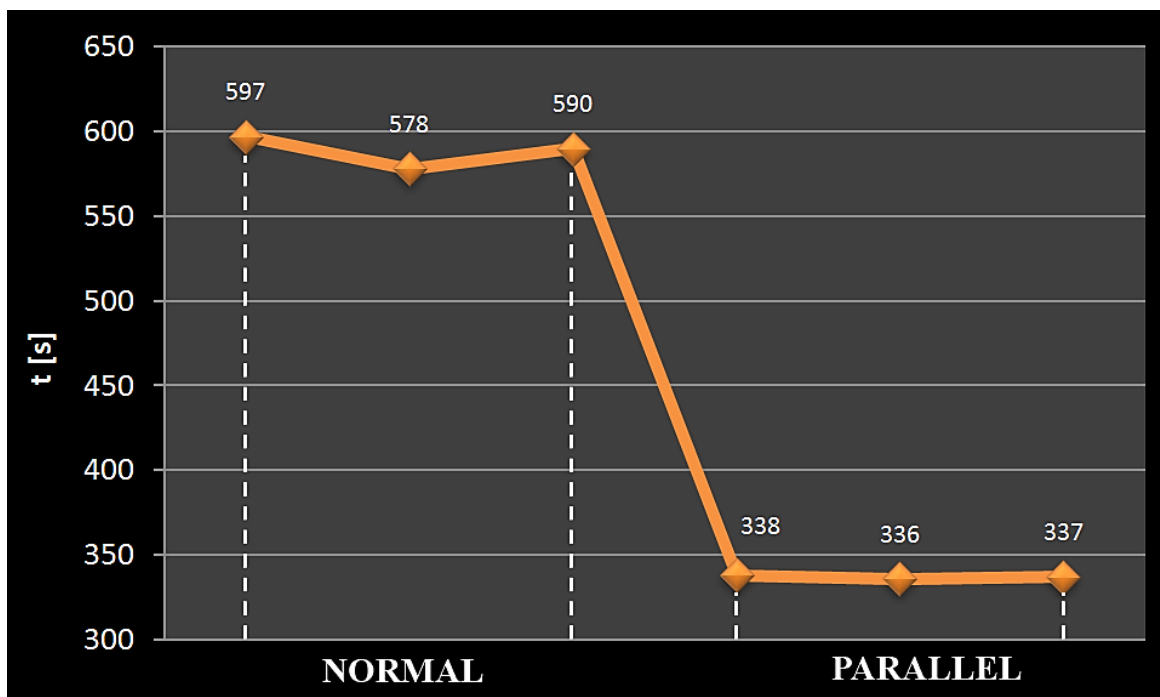
Obr. 27 Porovnanie GTIFF výstupov klasického a paralelného výpočtu

5. Zhodnotenie výsledkov a možnosti využitia

Analýzy viditeľnosti ako také majú veľmi široké možnosti využitia. Častokrát sú spájané v kontexte hodnotenia atraktivity a estetiky krajiny, ale rovnako tak sú využívané aj v armádnych, krajino-plánovacích či telekomunikačných činnostiach. Pri stále rozvíjajúcich sa možnostiach analýz viditeľnosti je potrebné aj ich optimalizovanie pre čo najväčšiu efektívnosť.

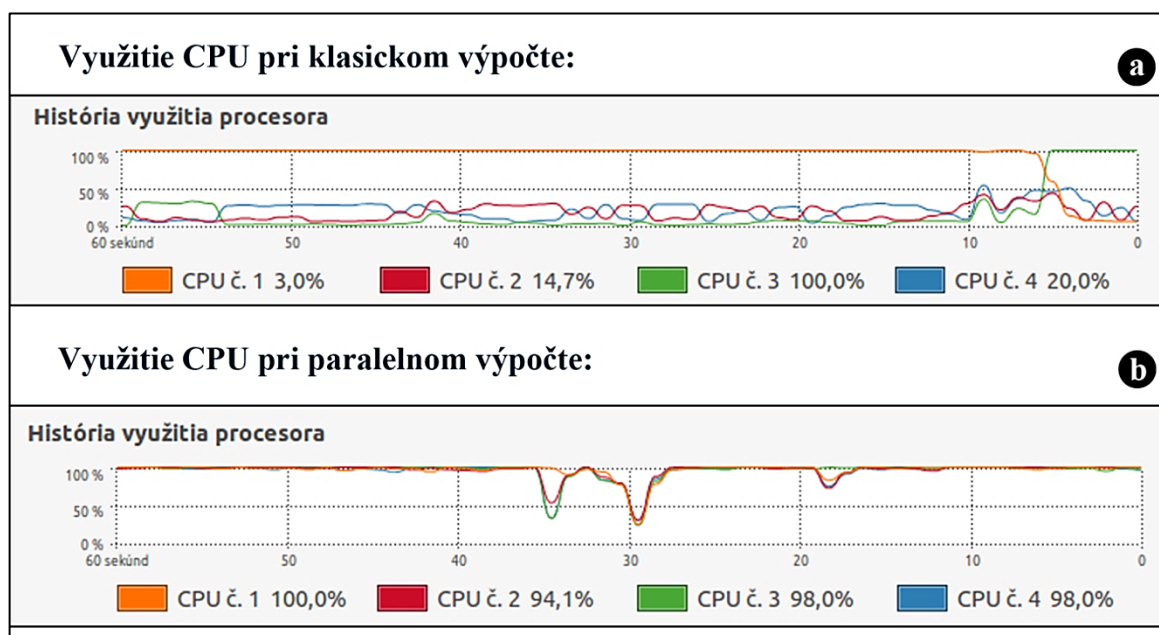
Snahou pri vytvorení týchto skriptov bolo zrýchlenie procesu výpočtu viditeľnosti pomocou paralelných výpočtov. Pre overenie zrýchlenia boli porovnávané časy výpočtov z testu algoritmu s časom výpočtu spustenej samotnej funkcie *r.viewshed* pre celú mapu.

Podmienky v oboch prípadoch boli identické. Boli použité rovnaké vrstvy v 25x25m rozlíšení s pozorovacím bodom vo výške 1,75 do vzdialenosti 50 km pre rovnaké súradnice. Výsledné mapy viditeľnosti boli taktiež identické (Obr. 27). Pre elimináciu neočakávaných faktorov a čo najpresnejšie výsledky prebehlo testovanie oboch nástrojov trikrát pri rovnakých podmienkach. Výsledné časy jednotlivých testov sú znázornené v grafe (Obr. 28).



Obr. 28 Graf zaznamenaných časov

Podľa aritmetických priemerov jednotlivých časov bola analýza viditeľnosti pomocou paralelných výpočtov o 4 minúty a 11 sekúnd rýchlejšia ako výpočet pomocou samotne spustenej funkcie *r.viewshed*. Výsledná rýchlosť sa teda zvýšila o **42,69 %**. Rozdiely sú jasne vidieť aj na obrázku č. 29a,b, zachytenom počas výpočtov a ktoré znázorňujú využitie jednotlivých CPU.



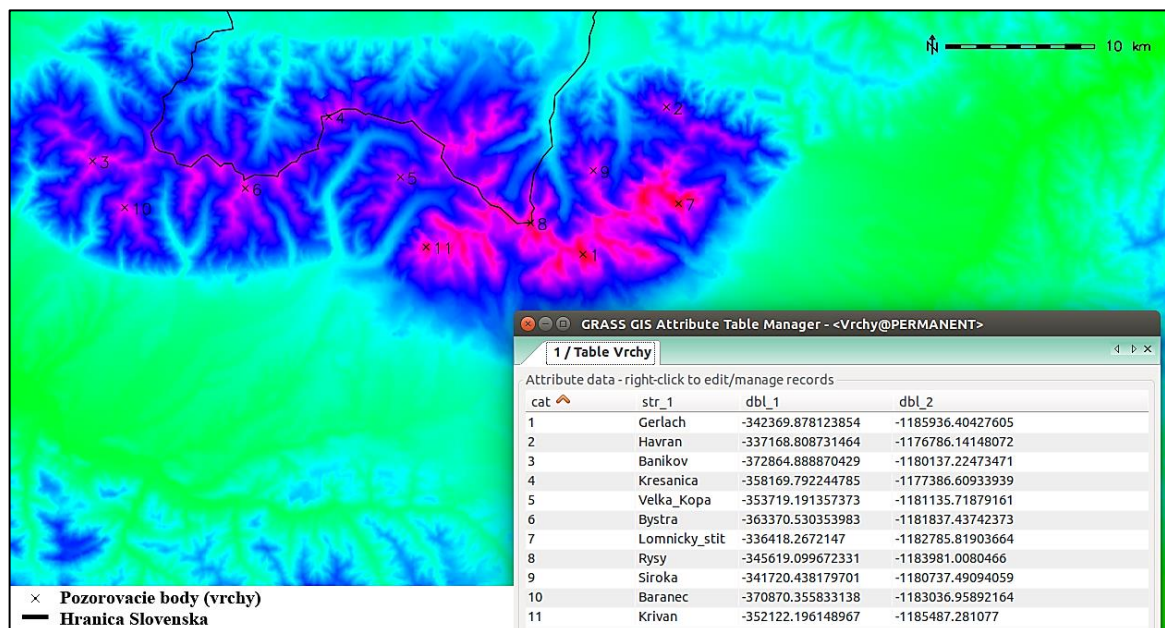
Obr. 29 Využitie CPU počas klasického a paralelného výpočtu

Obrázok č. 29a zobrazuje, ako je pri klasickom výpočte na celej mape využívaný len jeden procesor na 100%, nanajvýš dva procesory na 50%, zatiaľ čo na obrázku č. 29b pracujú všetky 4 procesory väčšinu času na 100%.

Takéto zrýchlenie však nie je konštantné ani maximálne. Rozdiely v rýchlosti sa budú meniť v závislosti od typu výpočtu (vzdialenosť, výška pozorovateľa, rozlíšenie DEM atď.), ale aj od typu rozdelenia mapy. Keďže testy prebiehali na štvorjadrovom počítači, skript bol prispôsobený na rozdelenie územia na štyri časti. Rozdelením územia na 6, 8, 16, 32... n počet častí pre počítanie na viacjadrových systémoch sa rýchlosť môže zvyšovať exponenciálne.

Okrem samotného testu algoritmu boli skripty spustené na viacerých počítačových stanicích s rôznym rozlíšením. Boli počítané viditeľnosti z rôznych Slovenských hradov a vrchov.

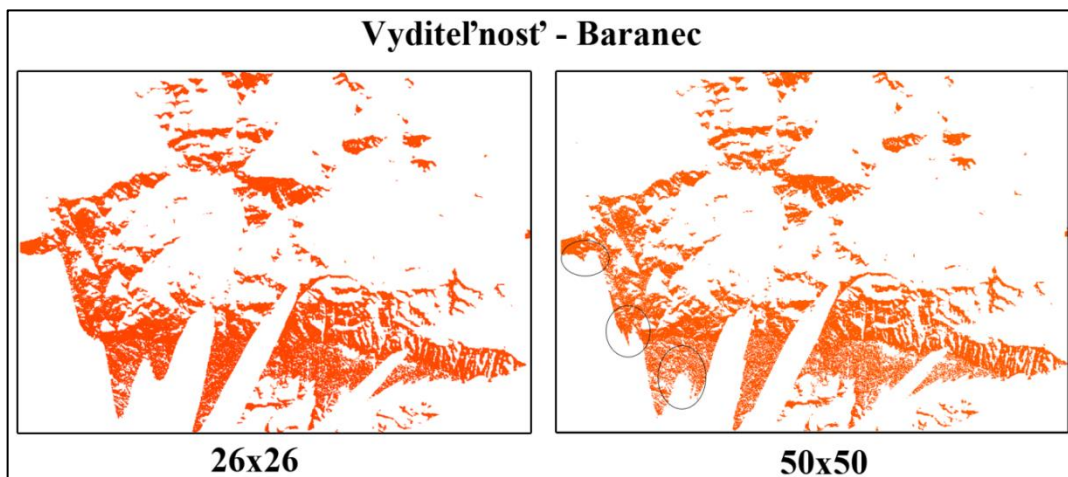
Príkladom môže byť analýza viditeľnosti vybraných tatranských vrchov s výškou nad 2000 metrov. Na základe jednotlivých geomorfologických celkov bolo vybraných 6 vrchov doplnených ďalšími 5-timi kvôli lepšiemu pokrytiu územia Tatier (Obr. 30).



Obr. 30 Vrchy pre ktoré bola počítaná viditeľnosť

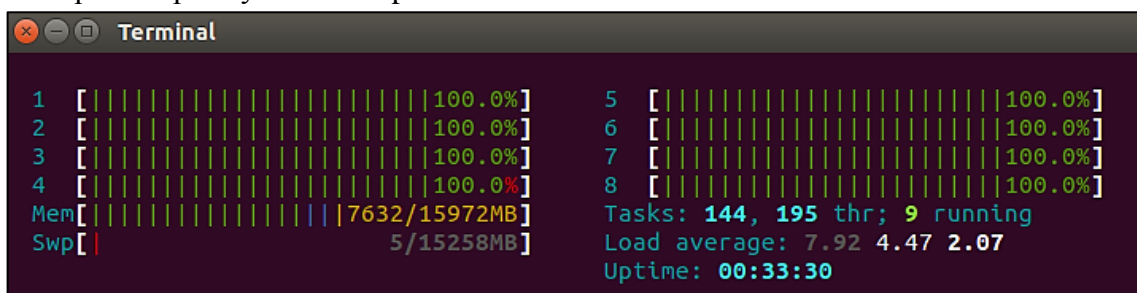
Výpočty prebiehali na 4 aj 8 jadrových počítačoch do vzdialenosti 200 a 250 km s 26x26m a 50x50m rozlíšením pri fixnej výške pozorovateľa 1,75m. Je nutné spomenúť že všetky tieto výpočty na výpočet viditeľnosti využívali nástroj *r.los* (GRASS GIS 6.4.4), ktorého algoritmus je znateľne pomalší ako algoritmus nástroja *r.viewshed*.

Vybrané výstupy:



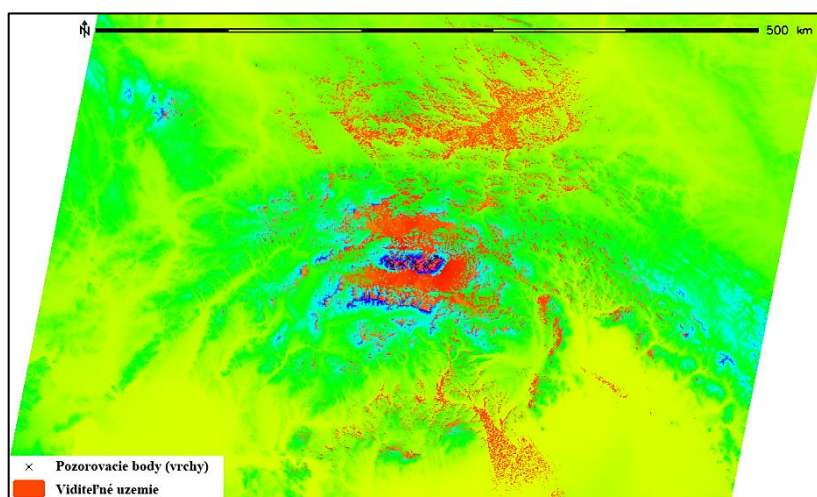
Obr. 31 Rozdiel výpočtu na vrstvách s 26m a 50m rozlíšením

Obrázok č. 31 znázorňuje rozdiel medzi 26x26m a 50x50m rozlíšením. Ako je možné vidieť, rozlíšenie siete ovplyvňuje výslednú mapu, ale tento vplyv nie je taký markantný ako napríklad pri Hydro.sh skripte.



Obr. 32 Využitie CPU na 8 jadrovom PC

Obrázok č. 32 zobrazuje využitie CPU pri výpočtoch na 8 jadrovom počítači.



Obr. 33 Viditeľnosť z 11 vybraných vrchov Slovenska

Obrázok č. 33 je mapa vzniknutá spojením viditeľností všetkých 11-ich vrchov do 250 km vzdialenosti, pri 50x50m rozlíšení.

3.5 Simulácia zaplavenia územia

Je možné simulovať vyliatie rieky v jej rôznych úsekoch ?

Aké veľké územie sa zaplaví pri pretrhnutí hrádze ?

Aké objekty sa zatopia ako prvé a akú škodu spôsobí záplava pri rôznych výškach ?

Nakoľko je takéto simulovanie presné ?

Povodne a záplavy sú jednou z najčastejších a zároveň najničivejších živelných pohrôm vo svete. Každoročne ohrozujú tisíce ľudských životov a spôsobujú veľké škody na majetkoch.

Európska smernica č. 2007/60/ES o hodnotení a manažmente povodňových rizík definuje povodeň ako dočasne zaplavené územie, ktoré zvyčajne nie je zaliate vodou.

Povodne a záplavy sú čím ďalej tým výraznejším determinujúcim faktorom v urbanizovanej krajine. Intenzita a frekvencia udalostí, ktoré dokážu vyvolať nadmerný odtok a následne povodeň, sa na Slovensku každým rokom zvyšuje. Výskyt povodní už nie je obmedzený len na flyšové oblasti, vyskytujú sa aj v oblastiach kotlín, nížin vyplnených fluviálnymi sedimentami, či v horských oblastiach tvorených vulkanickými horninami. Dá sa povedať, že povodeň môže nastať pozdĺž akéhokoľvek toku na Slovensku (Trizna, 2010).

1. Prístupy a opis simulácií zaplavovania území

Medzi najúčinnšie nástroje pri prevencii a zmierňovaní následkov povodní v povodňovom manažmente patria mapy povodňového rizika a inundačných území (Bačík, Mišík, Kučera, 2010).

K problematike mapovania povodní a inundačných území existuje viacero možných prístupov vychádzajúcich z hydraulických výpočtov, 1-D a 2-D hydrodynamického či numerického modelovania alebo menej presného štatistického regresného geomorfologického modelovania. Výber vhodnej metódy závisí najmä od typu požadovaného výsledku a od geografickej charakteristiky územia (Mišík, Kučera, 2010).

Pre možnosti simulácie odtokových procesov existuje niekoľko softvérových balíčkov ako napríklad WMS (Watershed modeling System), MIKE-11, MIKE-21, HEC-RAS, SMS(2-D) atď. Väčšina z týchto softvérov je komerčných a i pri ich schopnosti simulovať odtokové procesy v území, je ich najväčšou nevýhodou náročnosť na vstupné dáta a potrebnú prípravu, ktorá môže spravidla trvať niekoľko desiatok minút na jeden výpočet (Zorád, 2010).

Ak neberieme do úvahy všetky časopriestorové parametre ovplyvňujúce reálne zaplavovanie krajiny (intenzita zrážok, drsnosť terénu, typ využitia krajiny, typ a vlastnosti pôdy atď.), tak základnou metódou pri každom simulovaní záplav je porovnávanie výškových údajov DEM vrstvy daného územia zo zadanou výškou hladiny.

V našom skripte na toto simulovanie využívame GRASS GIS modul *r.lake*. Tento modul využíva funkciu “plávajúceho okna“ (moving window) o veľkosti 3x3 bunky, na nájdenie a vyplnenie všetkých buniek ktoré zodpovedajú trom základným kritériám (Nartiss, 2012):

- bunky majú nižšiu nadmorskú výšku ako je výška vodnej hladiny,
- bunky sú vzájomne prepojené,
- bunky nemajú NULL hodnotu alebo nie sú maskované.

Skript je založený na vytváraní záplavových máp s postupným zvyšovaním vodnej hladiny, pričom si užívateľ môže zvoliť počiatočné či konečné hodnoty a výšku, podľa ktorej sa bude hladina zvyšovať.

Skript rozoznáva typ vlozenej vrstvy s vodstvom, na základe čoho prebieha príprava údajov pre každý typ odlišne. Ak je vloženou vrstvou línia, tak skript ponúka možnosť ju rozsekať na kratšie, užívateľom navolené časti pre možnosť rátania zaplavenia v rôznych jej úsekoch. Ak po rozdelení línie zostane zvyšok, simulácia zaplavenia prebehne aj preň. Ak je vodstvo reprezentované bodmi a polygónmi je ponúknutá možnosť počítať zaplavenie pre každý objekt samostatne alebo pre celú vrstvu ako celok.

Skript obsahuje aj modul pre výpočet zaplavenia vybraných objektov. Modul pracuje na princípe štatistického prekrytia vlozenej vrstvy objektov (napr. obytné zóny, budovy, strategické objekty, atď.) spolu s výslednými záplavovými mapami. Pre čo najpresnejší výpočet prebieha štatistický prekryv vo vektorovej reprezentácii, ktorá umožňuje zachytiť aj čiastočné zatopenie.

2. Vstupné dáta

Základnými vrstvami vstupujúcimi do skriptu sú vektorová vrstva s vodstvom, DEM vrstva a vektorová vrstva reprezentujúca objekty.

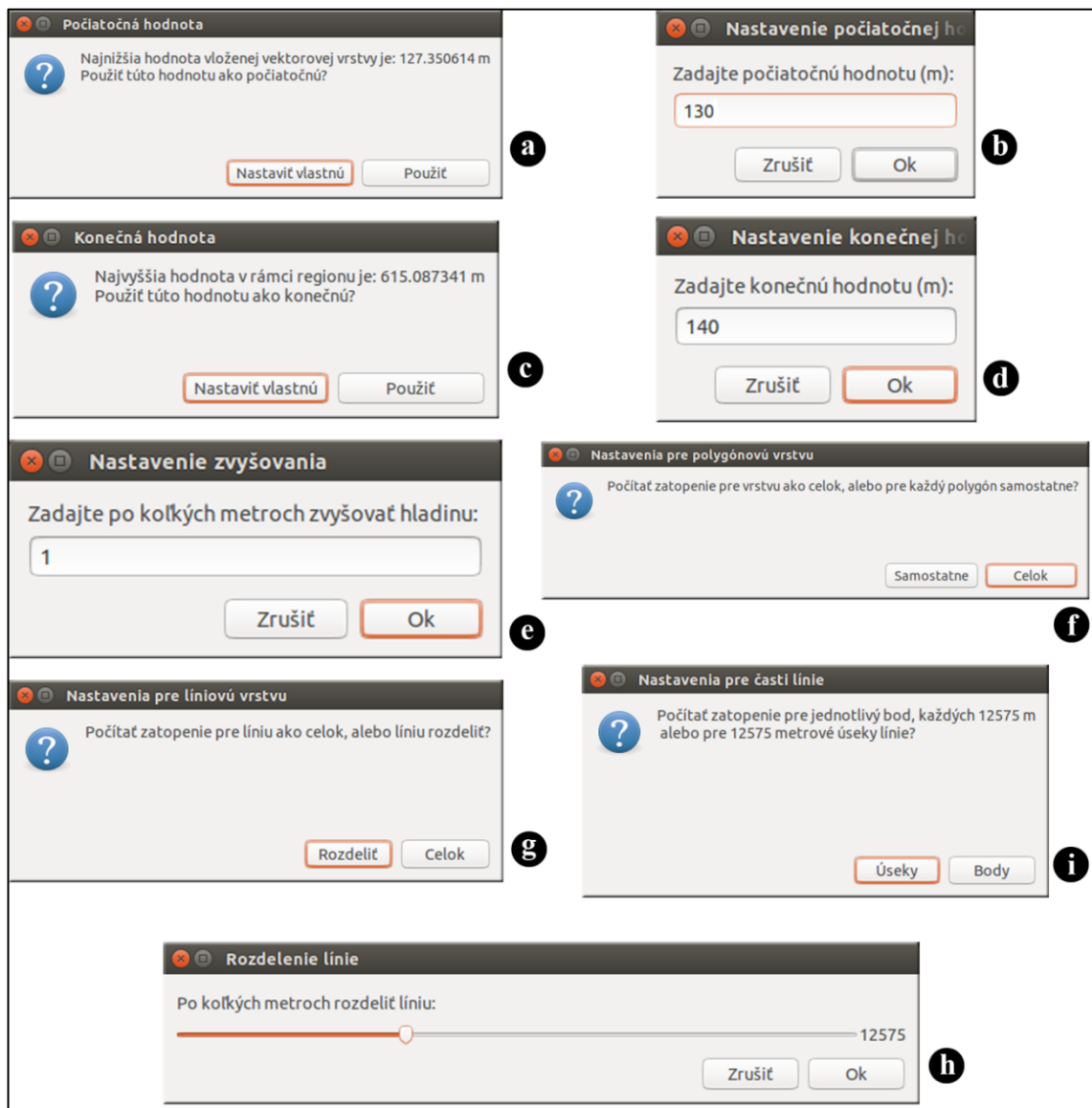
Voda vo vlozenej vrstve môže byť reprezentovaná bodmi, líniami, či polygónmi. Skript nevyžaduje aby vrstva tvorená viacerými bodmi (polygónmi) obsahovala stĺpec pre identifikáciu jednotlivých vodných prvkov. V prípade ak je zvolená možnosť počítať

zaplavenie pre každý z prvkov samostatne, tak ich identifikovanie prebieha na základe GRASS-om automaticky vytvoreného “cat“ stĺpca.

DEM vrstva je najkritickejšou zložkou každého softvéru na simulovanie zaplavenia územia. Pri tomto type výpočtov je pre čo najväčšiu presnosť simulovania, použitie DEM v čo najdetailnejšom rozlíšení dôležitejším faktorom ako pri ostatných typoch.

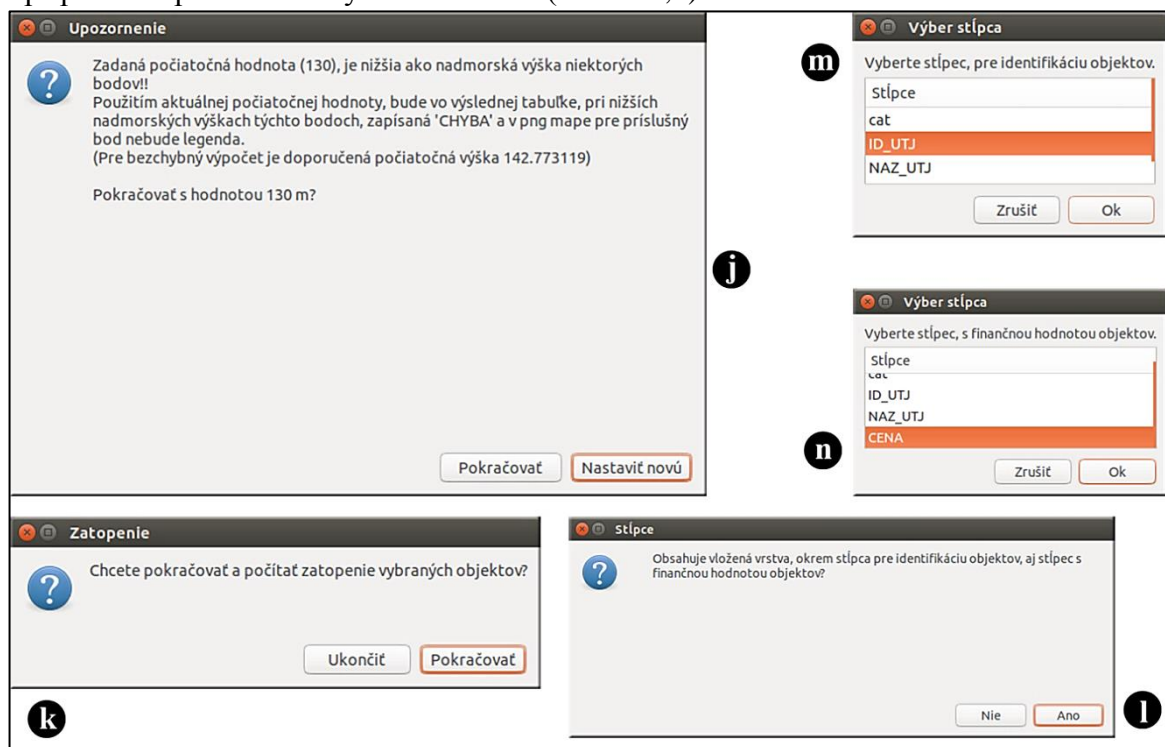
Pri využití možnosti vypočítať zatopenie vybraných objektov vstupuje do skriptu vektorová vrstva objektov. Objekty takisto môžu byť reprezentované bodmi, líniami, či polygónmi s rozdielom, že pri tejto vrstve sa očakáva numerický identifikátor jednotlivých objektov. Okrem identifikačného stĺpca môže vrstva niesť aj údaje s finančnými hodnotami objektov slúžiace na vyčíslenie škôd spôsobenými jednotlivými hladinami.

Zvyšné vstupy sú založené na báze interaktívnej komunikácie medzi skriptom a užívateľom prostredníctvom dialógových okien.



Obr. 34 Hydro.sh - Vstupy

Pri zadávaní počiatočných a konečných hodnôt (Obr. 34a-d) má užívateľ na výber, či počítať s hraničnými hodnotami určených na základe vložených vrstiev, alebo si hodnoty navoliť osobne. Ďalším vstupom je nastavenie o koľko sa bude hladina zvyšovať, kým nedosiahne konečnú hodnotu (Obr. 34e). V prípade bodovej či polygónovej vrstvy, užívateľ zadáva, či počítať zatopenie pre vrstvu ako celok, alebo pre každý bod (polygón) samostatne (Obr. 34f). V prípade línie pribúda možnosť rozdelenia na kratšie časti (Obr. 34g,h) a výber, či počítať zatopenie pre jednotlivé úseky alebo body (Obr. 34i). Ak je zadaná hladina menšia ako výška terénu, skript užívateľa upozorní (Obr. 35j) a poskytne možnosť zmeny počiatočnej hladiny pred začatím výpočtu. Pri spustení modulu zatopenia objektov (Obr. 35k) sú ďalšími vstupmi výber stĺpca s ID hodnotami (Obr. 35m) a prípadne stĺpca s finančnými hodnotami (Obr. 35l,n).



Obr. 35 Hydro.sh (modul zatopenia objektov) - Vstupy

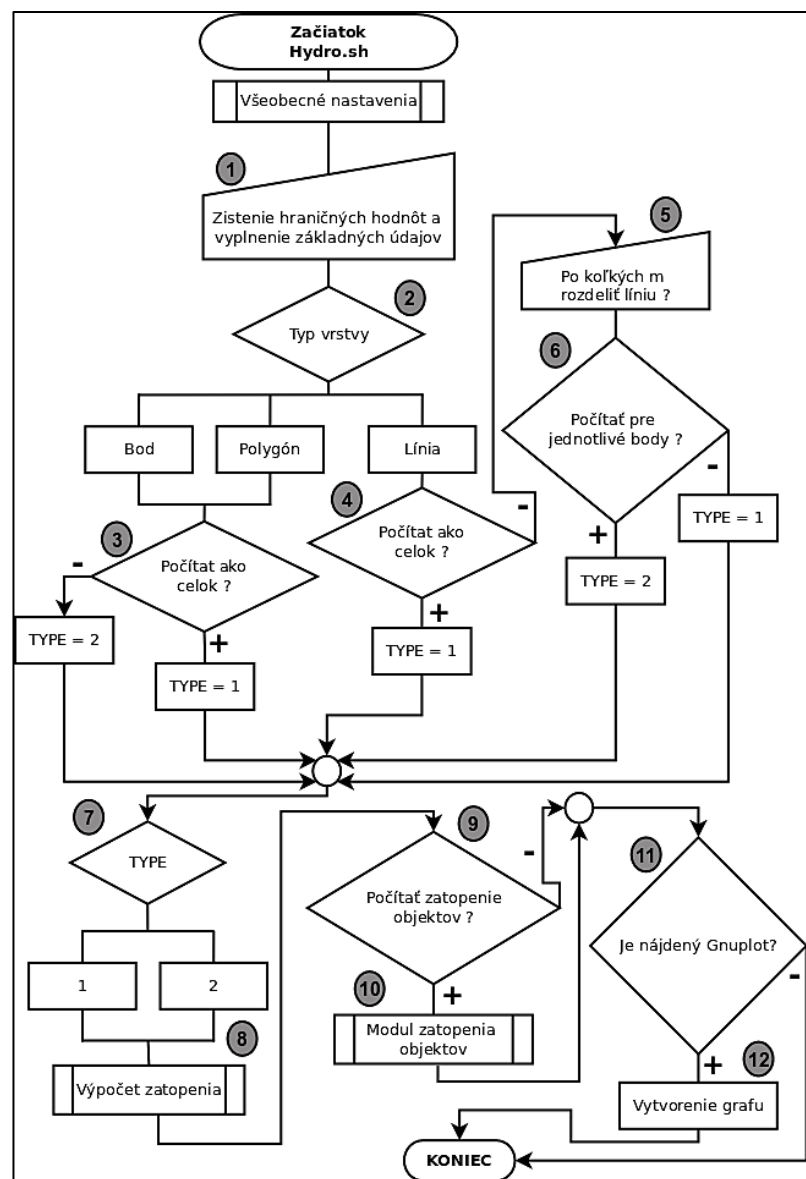
3. Opis algoritmu

Keďže tento skript obsahuje viacero funkcií, ktoré závisia od typu vlozenej vrstvy, je rozsiahlejší ako zvyšné skripty. Opis jeho algoritmu bol preto rozdelený do troch vývojových diagramov (VD. 10 - 12).

Prvým krokom je základná štatistika (*r.univar*) a štatistika prekryvu (*r.stats*) vložených vrstiev a určenie minimálnych a maximálnych hodnôt záujmového územia podľa nastaveného regiónu. Následne na to užívateľ vyplní základné údaje (počiatočná, konečná hodnota a zvyšovanie hladiny) (1). Keďže vložená vrstva s vodstvom môže byť

reprezentovaná akýmkoľvek z troch základných typov, skript ponúka niekoľko možností pre každý z nich (2). V prípade bodovej a polygónovej vrstvy je možnosť počítať zatopenie pre každý bod či polygón samostatne alebo rátať s vrstvou ako celkom (3). Na základe zvolenej možnosti sa prevedú osobitné úpravy pre každú vrstvu, pričom do premennej TYPE je zapísané akým spôsobom bude neskôr rávané zatopenie. V prípade línie si užívateľ vyberá, či počítať s vrstvou ako celkom alebo líniu rozdeliť na menšie časti (4). Ak je zvolené rozdelenie, tak skript najskôr zisťuje celkovú dĺžku línie a potom si užívateľ zvolí po koľkých metroch chce líniu deliť (5). Poslednou voľbou pri nastaveniach línie je otázka, či počítať zatopenie pre jednotlivé body vo zvolenej vzdialenosti alebo pre úseky (6). Nakoniec sa podľa akcie pripraví vstupná líniová vrstva (*v.split*; *v.to.points*; *v.report*) a zapíše sa premenná TYPE.

Na základe údajov zapísaných v premennej TYPE (7) prebehne samotný výpočet zatopenia (8) (VD. 11). Po výpočte zatopenia skript ďalej pokračuje otázkou, či má byť súčasťou výstupov aj výpočet zaplavených objektov (9) (10) (VD. 12). Ak nie, tak skript zisťuje, či sa v systéme nachádza program Gnuplot (11), a ak áno, tak prebehne vytvorenie grafu zaplavených území (12).



VD. 10 Vývojový diagram – Hydro.sh

Hydro (8)

Jadro výpočtu tvorí cyklus, ktorý sa opakuje pre daný počet bodov či polygónov (8A).

Ako prvé sa overuje, či je zadaná výška vyššia ako výška terénu (*r.stats*) pre aktuálne počítaný bod alebo polygón (8B). Ak nie je, skript užívateľa upozorní a vyzve k zmene počiatocnej hodnoty (8C). Ak nie je, skript užívateľa upozorní a vyzve k zmene počiatocnej hodnoty za odporúčanú výšku (8C).

Ak sa užívateľ rozhodne pre zmenu výšky (8D), tak skript opätovne ponúkne zadanie počiatocnej hodnoty (8E), ak nie, tak výpočet pokračuje s prvotne vloženou hodnotou.

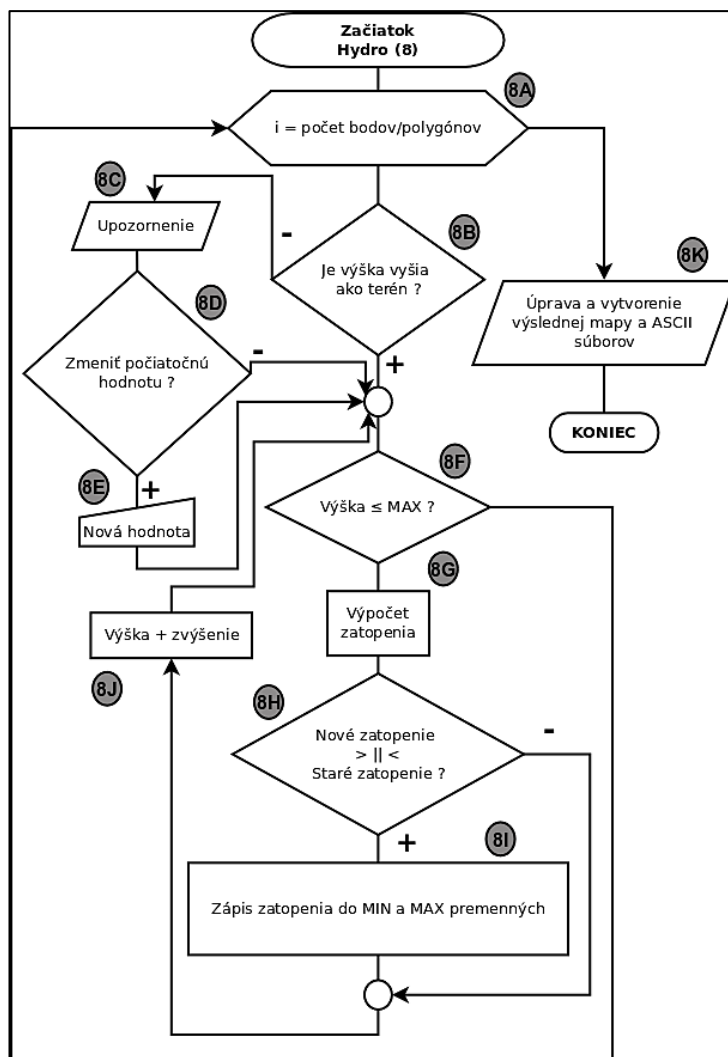
Ďalším krokom je cyklus s podmienkou opakujúci sa pokiaľ výška nedosiahne zadaného maxima (8F).

Na základe nazbieraných a pripravených údajov (*r.mapcalc*) prebieha výpočet zatopenia pre zvolené územie (*r.lake*) (8G).

Keďže jedným z výstupov je aj súbor z minimálnymi a maximálnymi zistenými hodnotami, skript obsahuje podmienku (8H), ktorá porovnáva výsledky veľkostí zatopených území a minimálne a maximálne hodnoty zapisuje do premenných (8I).

Každým cyklom sa hodnota výšky hladiny navýši o zadané zvýšenie (8J).

Posledným krokom je úprava vrstiev do výslednej podoby (*r.patch*; *r.mapcalc*; *r.colors*), export máp (*d.mon*) a vytvorenie ASCII súborov (8K).



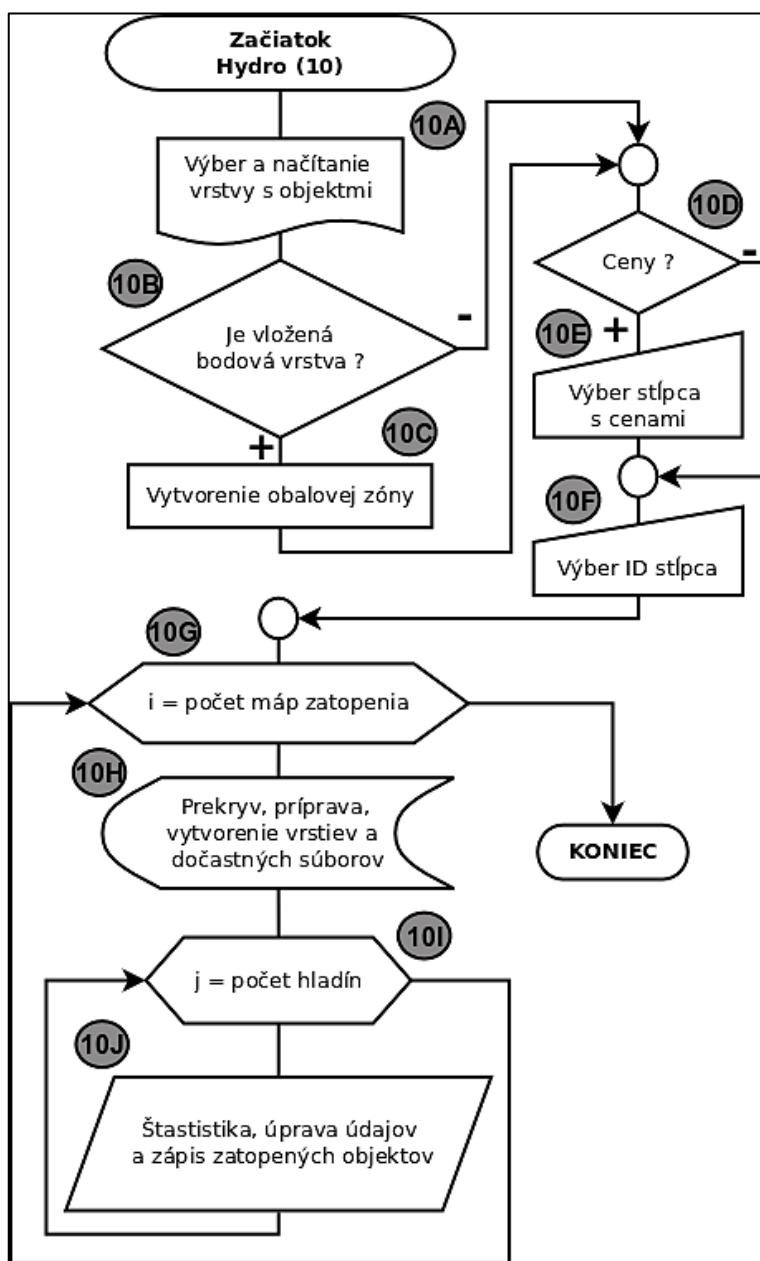
VD. 11 Vývojový diagram – Hydro (8)

Hydro (10)

Prvým krokom po spustení modulu zaplavenia objektov je výber a načítanie danej vrstvy objektov (*v.in.org*) (10A).

Keďže prekryv je možné robiť len s objektmi, ktoré majú rozmer, pri vložení bodovej vrstvy (10B) je každý bod prerobený na polygón pridaním obalovej zóny o priemere 0,1 m (*v.buffer*) (10C).

Skript následne od užívateľa zisťuje, či vrstva obsahuje stĺpec s finančnými hodnotami jednotlivých objektov (10D). Ak áno, užívateľ z listu možností postupne vyberie okrem stĺpca s ID hodnotami aj stĺpec s cenami (10E) (10F). Ak vrstva obsahuje len ID stĺpec, tak pri výslednom dokumente bude ku všetkým poškodeniam priradená hodnota "0".



VD. 12 Vývojový diagram – Hydro (10)

Hlavným krokom je cyklus opakujúci sa pre počet výsledných máp zatopenia (10G).

Pre výpočet zatopených objektov je potrebné upraviť a vytvoriť niektoré nové vrstvy (*r.to.vect*), z ktorých sú do dočasných súborov zaznamenané ich atribúty (*db.out.org*). V rámci tohto kroku je aj samotný prekryv vrstvy zatopenia s vrstvou objektov (*v.overlay*) (10H).

V druhom cykle (10I) prebieha posledná úprava údajov a zápis do výstupného súboru pre každú jednu hladinu výslednej vrstvy zatopenia (10J).

4. Test algoritmu a výstupy

Pre test algoritmu bola použitá DEM vrstva okresu Nitra v ASCII grid formáte s rozlíšením 10x10m. Ako vrstva reprezentujúca vodstvo bola zvolená líniová vrstva s riekou Nitrou. Bol testovaný aj modul zaplavenia objektov, do ktorého vstupovala polygónová vrstva predstavujúca intravilán vybraných sídiel nitrianskeho okresu.

Skript ponúka viacero rôznych možností výpočtu v závislosti od typu vlozenej vrstvy a nárokov užívateľa. Pre test boli zvolené nasledovné parametre:

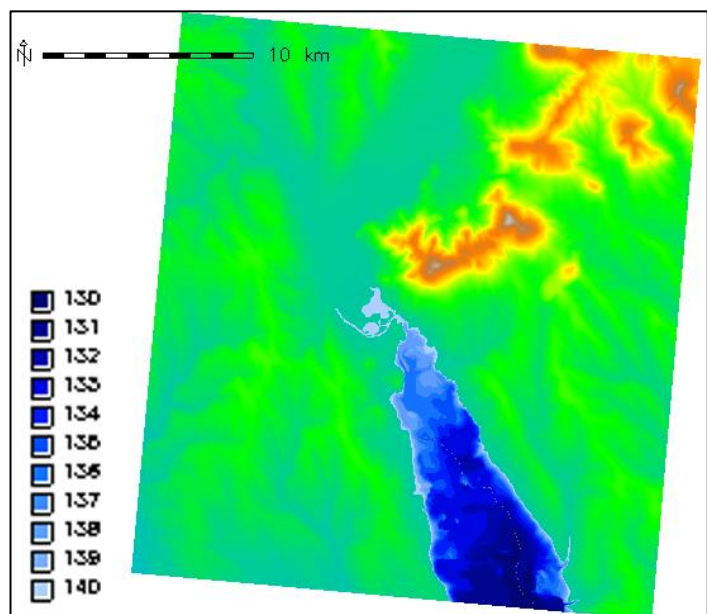
Počiatočná hodnota	130 m n. m.
Konečná hodnota	140 m n. m.
Zvyšovanie	1 m
Rozdeliť líniu ?	Áno
Úseky či body	12 575 m úseky
Počítať zatopenie objektov ?	Áno
Obsahuje finančný stĺpec ?	Áno

Keďže spád rieky Nitry v rámci výrezu je približne 27 m (najvyšší bod 154.35 m n. m., najnižší bod 127.35 m n. m.), našou snahou bolo simulovať zvýšenie hladiny v rôznych úsekoch jej toku. Vzhľadom na celkovú dĺžku (37725 m), bol tok rozdelený na tri 12575 m úseky, čoho výsledkom bola simulácia zatopenia pre horný, stredný a dolný úsek rieky.

Prvým výstupom sú mapy zaplavenia pre každý úsek (prípadne bod či polygón). Výsledné mapy sú pomenované v tvare map_1.png, map_2.png, ...map_n.png (Obr. 36).

Mapy sú v rozlíšení 640x480 pixlov a slúžia len ako náhľad. S dôvodu šetrenia miesta na disku sú jednotlivé rastrové vrstvy ponechané v spustenom mapsete.

Druhým výstupom je textový súbor "Vysledne_porovnanie.txt", ktorý pre každý úsek obsahuje informácie o maximálnej hĺbke, rozlohe zaplaveného územia a objemu vyliatej vody pre každú výšku hladiny samostatne (Obr. 37).



Obr. 36 PNG mapa zatopenia (map_3.png)

Názov: map_1 Úsek: 1-12575 m				
Výška_hladiny(m):		Maximálna_hĺbka(m):	Rozloha(m2):	Objem(m3):
130	CHYBA	CHYBA	CHYBA	
131	CHYBA	CHYBA	CHYBA	
132	CHYBA	CHYBA	CHYBA	
133	CHYBA	CHYBA	CHYBA	
134	CHYBA	CHYBA	CHYBA	
135	CHYBA	CHYBA	CHYBA	
136	CHYBA	CHYBA	CHYBA	
137	CHYBA	CHYBA	CHYBA	
138	CHYBA	CHYBA	CHYBA	
139	CHYBA	CHYBA	CHYBA	
140	CHYBA	CHYBA	CHYBA	

Názov: map_2 Úsek: 12575-25150 m				
Výška_hladiny(m):		Maximálna_hĺbka(m):	Rozloha(m2):	Objem(m3):
130	CHYBA	CHYBA	CHYBA	
131	CHYBA	CHYBA	CHYBA	
132	CHYBA	CHYBA	CHYBA	
133	CHYBA	CHYBA	CHYBA	
134	CHYBA	CHYBA	CHYBA	
135	3.757385	954800.000000	966275.012207	
136	7.310562	42576900.000000	150178894.070435	
137	8.310562	45271600.000000	194122060.873413	
138	9.310562	49192500.000000	241357593.768311	
139	10.310562	51227600.000000	291622838.233948	
140	11.310562	55199200.000000	344700003.717041	

Názov: map_3 Úsek: 25150-37725 m				
Výška_hladiny(m):		Maximálna_hĺbka(m):	Rozloha(m2):	Objem(m3):
130	1.104156	2914300.000000	892234.144592	
131	2.310562	9934800.000000	8772875.323486	
132	3.310562	15120200.000000	21669079.750061	
133	4.310562	27375400.000000	44750045.231628	
134	5.310562	31661200.000000	74455222.601318	
135	6.310562	36874700.000000	108252710.137939	
136	7.310562	42576900.000000	150178894.070435	
137	8.310562	45271600.000000	194122060.873413	
138	9.310562	49192500.000000	241357593.768311	
139	10.310562	51227600.000000	291622838.233948	
140	11.310562	55199200.000000	344700003.717041	

Obr. 37 Hydro.sh – Vysledne_porovnanie.txt

Na základe objemu vyliatej vody sú zaznamenávané minimálne a maximálne hodnoty, ktoré sú následne uložené do druhého textového výstupu "MIN_MAX.txt" (Obr. 38).

-----MINIMUM-----				
Meno_mapy	Výška_hladiny(m):	Maximálna_hĺbka(m):	Rozloha(m2):	Objem(m3):
map_3	130	1.104156	2914300.000000	892234.144592

-----MAXIMUM-----				
Meno_mapy	Výška_hladiny(m):	Maximálna_hĺbka(m):	Rozloha(m2):	Objem(m3):
map_2	140	11.310562	55199200.000000	344700003.717041
map_3	140	11.310562	55199200.000000	344700003.717041

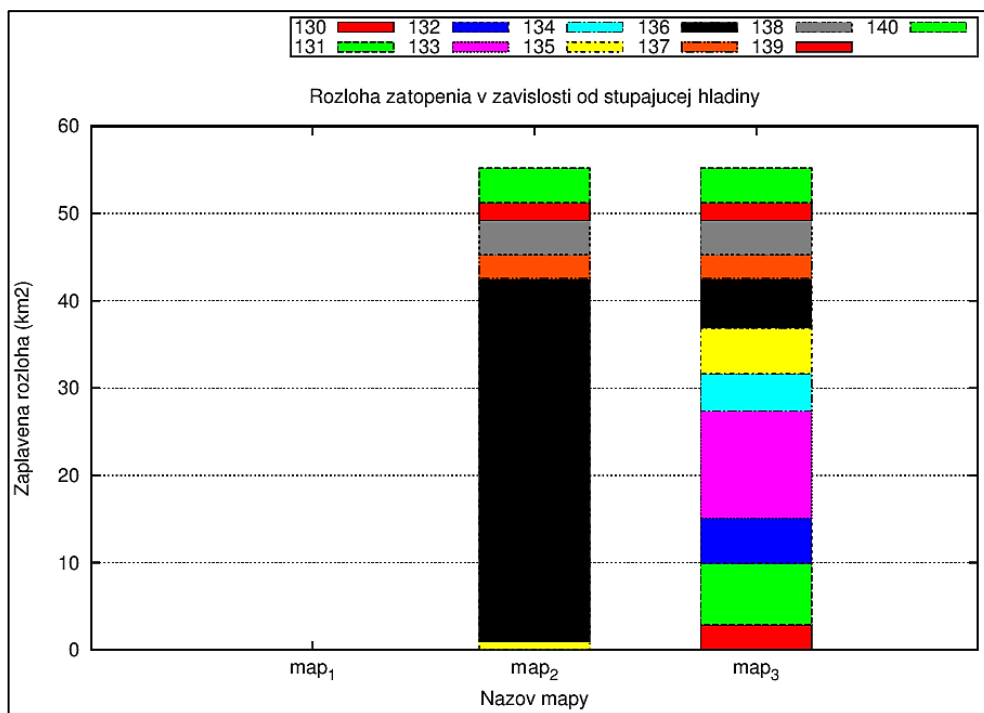
Obr. 38 Hydro.sh – MIN_MAX.txt

Keďže bola zvolená možnosť výpočtu zatopenia objektov, tretím textovým výstupom je zatopenie jednotlivých objektov pre každú mapu. Skript na základe zvoleného stĺpca s ID a finančnými hodnotami do súboru "Zatopenie.txt" zapisuje zatopené objekty pri jednotlivých výškach hladiny (Obr. 39).

<p>Zatopenie objektov pre: map_1</p> <p>Zatopenie objektov pre: map_2</p> <p>Výška hladiny: 136 m Zatopené objekty: 25; 27; 29; 30; 31; 42</p> <p>Finančné poškodenie zatopených objektov: 6000.06</p> <p>Výška hladiny: 137 m Zatopené objekty: 25; 27; 29; 30; 31; 39; 40; 42</p> <p>Finančné poškodenie zatopených objektov: 8000.08</p> <p>Výška hladiny: 138 m Zatopené objekty: 25; 27; 29; 30; 31; 39; 40; 42; 43; 44</p> <p>Finančné poškodenie zatopených objektov: 10000.10</p> <p>Výška hladiny: 139 m Zatopené objekty: 25; 27; 29; 30; 31; 39; 40; 42; 43; 44</p> <p>Finančné poškodenie zatopených objektov: 10000.10</p> <p>Výška hladiny: 140 m Zatopené objekty: 25; 27; 29; 30; 31; 36; 39; 40; 42; 43; 44</p> <p>Finančné poškodenie zatopených objektov: 11000.11</p>	<p>Výška hladiny: 130 m Zatopené objekty: 30</p> <p>Finančné poškodenie zatopených objektov: 1000.01</p> <p>Výška hladiny: 131 m Zatopené objekty: 27; 30; 31</p> <p>Finančné poškodenie zatopených objektov: 3000.03</p> <p>Výška hladiny: 132 m Zatopené objekty: 25; 27; 30; 31</p> <p>Finančné poškodenie zatopených objektov: 4000.04</p> <p>Výška hladiny: 133 m Zatopené objekty: 25; 27; 29; 30; 31; 42</p> <p>Finančné poškodenie zatopených objektov: 6000.06</p> <p>Výška hladiny: 134 m Zatopené objekty: 25; 27; 29; 30; 31; 42</p> <p>Finančné poškodenie zatopených objektov: 6000.06</p> <p>Výška hladiny: 135 m Zatopené objekty: 25; 27; 29; 30; 31; 42</p> <p>Finančné poškodenie zatopených objektov: 6000.06</p> <p>Výška hladiny: 136 m Zatopené objekty: 25; 27; 29; 30; 31; 42</p> <p>Finančné poškodenie zatopených objektov: 6000.06</p> <p>Výška hladiny: 137 m Zatopené objekty: 25; 27; 29; 30; 31; 39; 40; 42</p> <p>Finančné poškodenie zatopených objektov: 8000.08</p> <p>Výška hladiny: 138 m Zatopené objekty: 25; 27; 29; 30; 31; 39; 40; 42; 43; 44</p> <p>Finančné poškodenie zatopených objektov: 10000.10</p> <p>Výška hladiny: 139 m Zatopené objekty: 25; 27; 29; 30; 31; 39; 40; 42; 43; 44</p> <p>Finančné poškodenie zatopených objektov: 10000.10</p> <p>Výška hladiny: 140 m Zatopené objekty: 25; 27; 29; 30; 31; 36; 39; 40; 42; 43; 44</p> <p>Finančné poškodenie zatopených objektov: 11000.11</p>
<p>a</p>	<p>b</p>

Obr. 39 Hydro.sh – Zatopenie.txt

Ak v rámci systému, v ktorom je skript spustený, je nainštalovaný aj vykresľovací softvér Gnuplot, tak výsledkom bude aj graf porovnávajúci rozlohy zatopených častí pri jednotlivých výškach hladiny (Obr. 40). Rozlohy vo výslednom grafe sú kvôli lepšej prehľadnosti premenené na km².



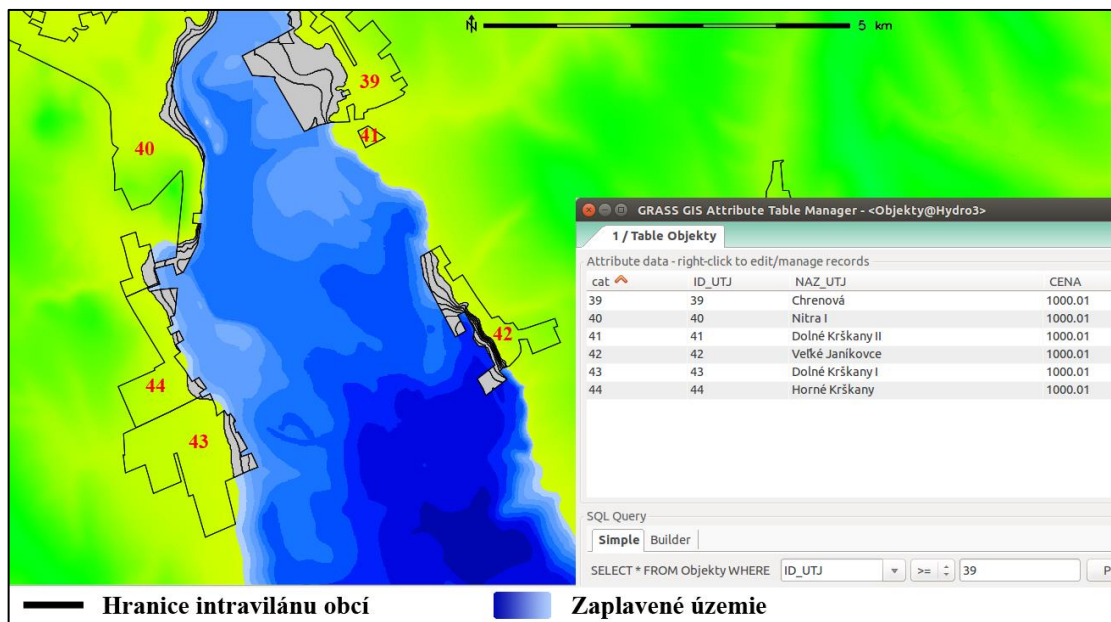
Obr. 40 Výsledný graf rozlôh zaplavených území

5. Zhodnotenie výsledkov a možnosti využitia

Pred zhodnotením výsledkov treba zdôrazniť, že test bol zameraný čisto na funkčnosť a ukážku algoritmu a výsledky z neho by nemali byť brané ako reálne simulácie zatopenia územia riekou Nitra. Pre simuláciu reálneho zatopenia sú potrebné čo možno najpresnejšie vstupné údaje a starostlivo zvolené hodnoty.

Z obrázkov č. 37, 39 a 40 je vidieť, že pre horný úsek toku neboli vypočítané žiadne hodnoty. Ako už bolo písané, vo vyselektovanom výreze Nitry je rozdiel medzi najvyšším a najnižším bodom približne 27 metrov. Keďže, aj napriek varovaniu zo strany skriptu (Obr. 35j), nebola zmenená počiatková výška hladiny na odporúčanú (142,773119), tak v rámci horného úseku (map₁) prebehol každý výpočet pod výškou terénu a tým pádom bola do výsledného porovnania zapísaná "CHYBA". Pri strednom úseku rieky (map₂), bola výška počiatkovej hladiny aj napriek upozorneniu takisto ponechaná, čím prebehol výpočet len na šiestich výškach a vo výslednej mape chýbala mierka. Pri dolnom úseku (map₃) bola odporúčaná počiatková výška hladiny dodržaná a výpočet prebehol bez komplikácií pre každú výšku (Obr. 36).

Keďže prekryv zatopených objektov prebieha vo vektorovej reprezentácii, skript zachytí a zaznamená aj okrajovo zatopené objekty. Z výsledného dokumentu (Obr. 39) a obrázku č. 41 je jasne vidieť, že na zaznamenanie objektu stačí aj čiastočne zatopenie.



Obr. 41 Ukážka čiastočného zatopenia objektov

Simulácie inundačných území sú účinným prostriedkom pre rozhodovanie, plánovanie a manažment protipovodňovej ochrany. Skript umožňuje vytvorenie simulácií záplav pre situácie ako napríklad pretrhnutie hrádze v určitom bode, vyliatie rieky pre jednotlivé jej úseky, či výpočet zaplavenia pre rôzne vodné plochy spolu s možnosťou zaplavovania vybraných objektov. Vďaka automatizácii môže užívateľ simulovať záplavy pre viacero bodov alebo plôch, a to buď pre každý samostatne alebo ako celok. Avšak miera korektnosti vytvorených simulácií úzko závisí s kvalitou a presnosťou vstupných dát. Existuje viacero príspevkov a monografií zaoberajúcich sa presnosťou a chybovosťou DEM. Ako uvádza Podhorányi et. al., (2011) v príspevku venovanom hodnoteniu presnosti DEM pri modelovaní povodní, tak „Hydrodynamické modely vykazujú výraznú citlivosť na vstupné dáta a už pri minimálnych nepresnostiach sa vo výsledkoch zobrazujú veľké nepresnosti.“

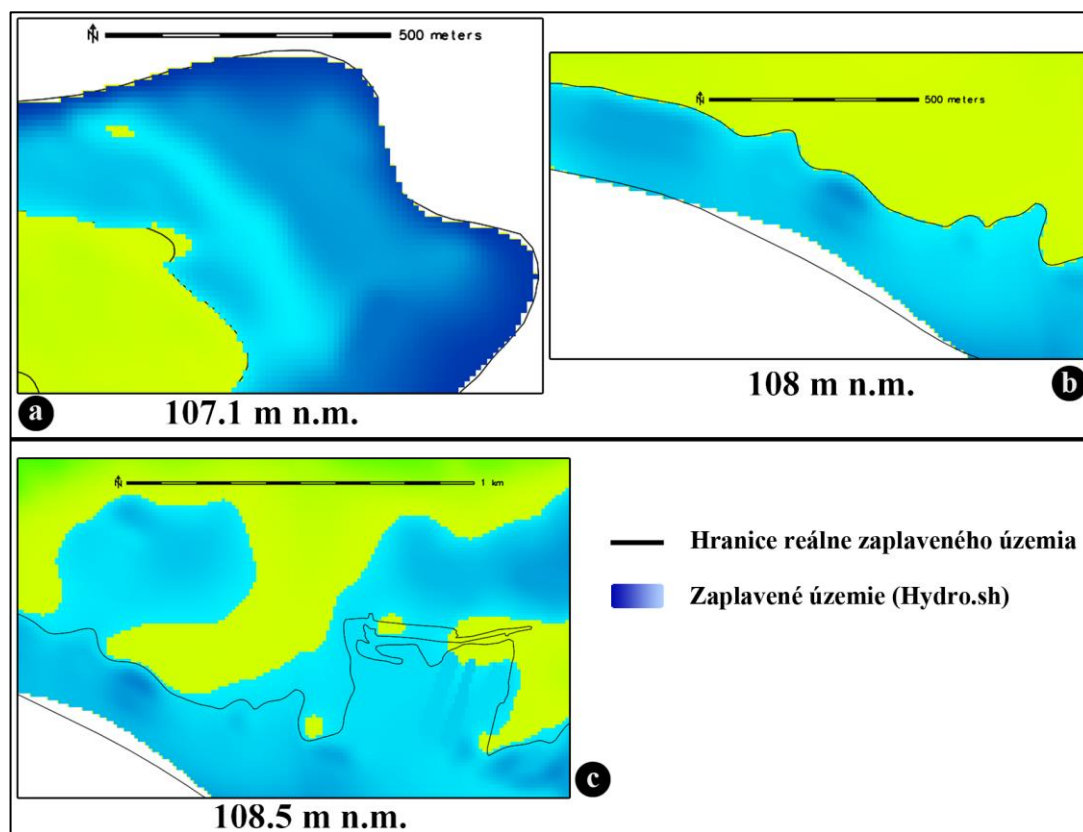
Vo všeobecnosti možno povedať, že pre čo najrealistickejšie simulovanie záplav je vhodné použiť DEM v rozlíšení 5x5m a vyššie.

Naším druhým testom bolo porovnanie simulovaných záplav zo skriptu spolu s vrstvou zachycujúcou reálne zaplavené územie. Porovnanie prebiehalo v rámci katastrálneho územia Štúrovo na sútoku riek Dunaj a Ipeľ. Vstupnými vrstvami bola DEM s rozlíšením 10x10m a polygónová vrstva s výrezom oboch riek. Porovnávajúca vrstva bola vytvorená priamym GPS meraním zaplavených častí území v teréne Mgr. Henrichom Grežom, PhD.

Zvolené údaje:

Výpočet zatopenia	Počítať pre celú vrstvu
Počiatková výška	107
Konečná výška	110
Zvyšovanie	0,1 m
Výpočet zatopených objektov	Nie

Výsledné porovnanie ukázalo, že simulovanie pri určitých výškach bolo na niektorých častiach územia skutočne blízke reálne zameraným záplavam (Obr. 42a,b). Avšak niektoré časti boli zaplavené viac alebo menej (Obr. 42c).



Obr. 42 Porovnanie simulovaného a reálneho zaplavenia územia - Štúrovo

Ak by cieľom testu bolo vytvoriť mapu simulácie čo najpodobnejšiu porovnávacej vrstve, bolo by nutné splnenie týchto kritérií:

- zlepšenie kvality vstupného DEM alebo zámena za DEM vo vyššom rozlíšení,
- údaje, o koľko bola hladina na oboch tokoch zvýšená.

ZÁVER

Výstupom práce je deväť skriptov s využitím v rôznych oblastiach ekológie a environmentalistiky rozdelených do štyroch kategórií podľa hlavných možností využitia. Jednotlivé skripty sú vytvorené v BASH skriptovacím jazyku s využitím GRASS GIS a jeho analytických a štatistických nástrojov.

Opisy algoritmov, vstupy, testovania, výstupy, zhodnotenia, porovnania a možnosti využitia sú pre každú kategóriu skriptov charakterizované samostatne v kapitole 3 (Výsledky a diskusia). Pre zhrnutie najdôležitejších údajov zistených počas testovania jednotlivých skriptov možno povedať že:

- klasické 2-D metódy výpočtu rozlohy vykazujú značné rozdiely oproti reálnej 3-D rozlohe územia. I keď veľkosť rozdielu vo výsledkoch závisí najmä od veľkosti a členitosti samotného územia, rozdiely v našich testoch, sa pohybujú od 0.5 do 8% oproti vyrátanej 2-D rozlohe, čo na území Slovenska činí približne 1162.5426 km²,
- pri správnom využití automatizácie je možné vykonávať analýzy a porovnania aj niekoľko tisíc riadkových databáz v akejkolvek špecifickej forme za pár sekúnd a prepojiť ich s priestorovými údajmi, čím je sa môže ušetriť množstvo času, ako aj finančných zdrojov,
- výpočet viditeľnosti má široké možnosti využitia pri mnohých činnostiach. Vzhľadom k tomu, že ide o časovo náročný proces, sa paralelné výpočty ukázali ako účinný spôsob jeho zrýchlenia. Pri testoch na štvorjadrových procesoroch sme dosiahli zvýšenie rýchlosti o 42,69 %, pričom samotná rýchlosť rátania paralelnými výpočtami úmerne rastie s množstvom použitých jadier,
- na základe porovnania reálne zameraných povodňových máp s výsledkom testu prevedenom na digitálnom modeli reliéfu s rozlíšením 10x10m je za použitia vhodných vstupných údajov možné pomerne efektívne simulovať zaplavenie územia, pričom veľkou výhodou oproti ostatným softvérom ostáva možnosť rátať zatopenie vybraných objektov v krajine,
- najdôležitejším faktorom pre dosiahnutie čo najpresnejších výsledkov je kvalita vstupných údajov a to najmä DEM.

Naším zámerom je naďalej pokračovať vo vývoji, vylepšovaní a vytváraní nových verzií týchto skriptov, vzhľadom na možnosť ďalšej spolupráce napríklad s doc. Mgr. Ivanom Balážom, PhD., doc. PhDr. Petrom Ivaničom, PhD., či tímom Ing. Viera Vanková, PhD. a RNDr. Peter Petluš, PhD.

Vzhľadom k tomu, že vytvorené skripty sú novými alebo vylepšenými nástrojmi pre podporu výskumu krajiny, dúfame, že si svoje miesto nájdu aj pri využití v praxi.

ZOZNAM POUŽITEJ LITERATÚRY

BAČÍK, M. – MIŠÍK, M. – KUČERA, M. 2010. *Využitie matematických modelov a nástrojov hydroinformatiky v manažmente povodňových rizík.* [online]. Bratislava, DHI SLOVAKIA, s.r.o. 9 s. [cit. 3.2.2015]. Dostupné na internete:

<http://www2.dhi.sk/publikacie/WRH-ManazementPovodnovychRizik-Sk.pdf>

BALÁŽ, I. et al. 2014a. *Priestorové interakcie piskorov v lesnom ekosystéme Tatier.* In Zborník príspevkov z vedeckého kongresu „Zoológia 2014“, 19. Feriencove dni. Prešov : Prešovská univerzita v Prešove, 2014. ISBN 978-80-555-1140-5.

BALÁŽ, I. et al. 2014b. *Effect of the environmental factors on the small mammals' spatial activity under the forest conditions of Tatras Mts.* In 14th Rodens et Spatium Conference on Rodent Biology. Lisabon, 2014.

BELIANSKÝ, P. 2015. *Revitalizácia súčasného stavu MVN : diplomová práca,* 2015, in press

BERNHARDSEN, T. 1992. *Geographic Information Systems An Introduction.* 2. Vyd. New York : John Wiley & Sons, 2002. 448 p. ISBN 978-0-471-41968-6.

BLAKE, L. 2007. *Spatial Relationships In GIS-An Introduction..* In OSGeo Journal, Canada, 2007. 1(1), ISSN 1994-1897

BŘEHOVSKÝ, M. – JEDLIČKA, K. 2005. *Úvod do geografických informačních systémů.* Plzeň : Západočeská univerzita v Plzni, 2005. 116 s.

BURKHOLDER, E. F. 2001. *Spatial data, coordinate systems, and the science of measurement.* In Journal of surveying engineering., United States, 2001. 127(4), ISSN: 1943-5428 p. 143-156.

CÂMARA, G. et al. 2005. *Towards a generalized map algebra: principles and data types.* In VII Simpósio Brasileiro de Geoinformática. Brasil, 2005. ISBN 85-17-00027-7, p. 66-81.

DE SILVA, F. N. – EGGLESE, R. W. 2000. *Integrating simulation modelling and GIS: spatial decision support systems for evacuation planning.* In Journal of the Operational Research Society, 2000, p. 423-430.

- EASTMAN, R. J. et al. 1993.** *The GIS Handbook*. Massachusetts : Clark University, 1993. 68 p.
- EGENHOFER, M. et al. 1989.** *A formal definition of binary topological relationships*. In *Computer Science*, 1989, vol. 367, p. 457-472.
- ESCOBAR, F. et al. 1999.** *Introduction to GIS*. Melbourne : Department of Geomatics, The University of Melbourne, 1999. 13 p.
- ESRI, 2012.** *What is GIS?*. New York Street, Redlands, California, 2012. 64 s.
- ESTER, M. et al. 2000.** *Spatial Data Mining: Database Primitives, Algorithms and Efficient DBMS Support*. In *Data Mining and Knowledge Discovery*, 2000, 4.2-3, p. 193-216. ISSN: 1384-5810
- FERREIRA, C. R. et al. 2014.** *A parallel algorithm for viewshed computation on grid terrains*. In *Journal of Information and Data Management*, 5(2), 2014, p. 171. ISSN 2178-7107
- FISHER, P. F. 1996.** *Extending the applicability of viewsheds in landscape planning*. In *Photogrammetric Engineering and Remote Sensing*, 62(11), 1996, p. 1297-1302.
- GOODCHILD, M. F. 2001.** *Geographic information system*. In *Encyclopedia of Global Change*, New York : Oxford University Press vol, 2001, 6 p.
- HOFIERKA, J. 2003.** *Geografické informačné systémy a diaľkový prieskum Zeme*. Prešov : Prešovská univerzita v Prešove, 2003. 116 s. ISBN 80 – 8068 – 219 - 4.
- HRUBÝ, M. 2006.** *Geografické informačné systémy*. Brno : FIT VUT Brno, 2006. 98 s.
- HUISMAN, O. - ROLF, A. 2009.** *Principles of Geographic Information Systems*. 3. vyd. Enschede : The International Institute for Geo-Information Science and Earth Observation, 2009. 540 p. ISBN 978–90–6164–269–5.
- CHAO, F. et al. 2011.** *Parallel algorithm for viewshed analysis on a modern GPU*. In *International Journal of Digital Earth*, 4(6), 2011, p. 471-486. ISSN 1753-8955
- JACKSON, M. J. – WOODSFORD, P. A. 1991.** *GIS data capture hardware and software*. In *Geographical information systems*, vol. 1, 1991, p. 239-249.

- JAKUBCOVÁ, A. – GREŽO, H. – PETROVIČ, F. 2014.** *A Proposal for the tourism attraction in the south-west part of Slovak agricultural landscape.* In Public recreation and landscape protection – with man hand in hand? : Conference proceeding. Brno : Mendel University in Brno, 2014. ISBN 978-80-7375-952-0.
- JAKUBÍK, J. 2010.** *Základy kartografie a topografie.* Banská Bystrica : Univerzita Mateja Bela, 2010. 144 s. ISBN 9788055700656.
- KAISOVÁ, D. 2015.** *Vývoj rozptýleného osídlenia v mestách Krásno nad Kysucou, Turzovka a Čadca* : diplomová práca, 2015, 91 s. in press
- KIM, Y. H. – RANA, S. – WISE, S. 2004.** *Exploring multiple viewshed analysis using terrain features and optimisation techniques.* In Computers & Geosciences, 30(9), 2004, p. 1019-1032.
- KOLÁŘ, I. – POSPÍŠILOVÁ, L. 2007.** *Diferenciální geometrie křivek a ploch.* Brno : Přírodovědecká fakulta, Masarykova univerzita v Brne, 2007. 193 p.
- KOLEKTÍV AUTOROV, 2012.** *Nasazení Open Source Software: výhody a nevýhody.* Praha : Institut pro informační společnost, o.s., 2012. 108 s.
- KOMÁRKOVÁ, J. – KOPÁČKOVÁ, H. 2008.** *Geografické informační systémy.* 2. vyd. Pardubice : Fakulta ekonomicko-správní, Univerzita Pardubice, 2008. 55 s. ISBN 978 - 80 - 7395 - 120 – 7.
- KOREŇ, M. 1995.** *Svet priestorových informácií.* In Geoinfo, 1995, 2(1), p. 25-29.
- KRÁSA, J. et al. 2006.** *Geografické informační systémy: jako podklad rozhodovacího procesu, zejména pro úkoly krajinného inženýrství.* Praha : Fakulta stavební ČVUT v Praze, 2006. 85 s.
- KUSEDOVÁ, D. 2003.** *Geografické informačné systémy a humánna geografia — vybrané teoreticko–metodologické a aplikačné aspekty.* In Acta Facultatis Rerum Naturalium Universitatis Comenianae (AFRNUC), Geographica, vol. 44, ISSN 80-223-1886-8, 2003, p. 89-140.
- LONGLEY, P. et al. 2005.** *Geographical Information Systems and Science.* 2. vyd. England : John Wiley & Sons, 2005. 517 s. ISBN 0-470-87001-X (PB).

- MAGUIRE, D. J. 1991.** *An Overview and Definition of GIS*. In Geographical Information Systems: Principles and Applications, 1991, 1, p. 9-20. ISBN 0-582-05661-5
- MAJERNÍK, J. 2008.** *Základy informatiky*. Košice : Aprilla, s.r.o., 2008. 154 s. ISBN 978-80-89346-03-5.
- MALOY, A. M. – DEAN, D. J. 2001.** *An Accuracy Assessment of Various GIS-Based Viewshed Delineation Techniques*. In Photogrammetric Engineering and Remote Sensing, 67(11), 2001, p. 1293-1298.
- MIKLÓS, L. – ŠPINEROVÁ, A. 2010.** *Priestorová diverzita krajiny*. Harmanec : VKÚ, a. s., 2010. 68 s. ISBN 978-80-89585-16-8.
- MILLER, M. L. 2011.** *Analysis of Viewshed Accuracy with Variable Resolution LIDAR Digital Surface Models and Photogrammetrically-Derived Digital Elevation Models* : diplomová práca. Virginia: Virginia Polytechnic Institute and State University, 2011. 98 s.
- MIŠÍK, M. – KUČERA, M. 2010.** *Rôzne prístupy pri definovaní potenciálneho inundacného územia*. [online]. Bratislava, DHI SLOVAKIA, s.r.o. 9 s. [cit. 3.2.2015]. Dostupné na internete: http://www2.dhi.sk/publikacie/Podbanske07_potenc_zona_DHI.pdf
- MITÁŠOVÁ, H. – HOFIERKA, J. 1993.** *Interpolation by Regularized Spline with Tension: II. Application to terrain modeling and surface geometry analysis*. In Mathematical Geology, 1993, vol. 25, p. 657-669. ISSN 1573-8868
- NARANG, R. 2006.** *Database Management Systems*. India : Prentice-Hall Of India, 2006. 432 p. ISBN 9788120326453.
- OSTERMAN, A. 2012.** *Implementation of the r. cuda. los module in the open source GRASS GIS by using parallel computation on the NVIDIA CUDA graphic cards*. In Elektrotehniski Vestnik, 79(1-2), 2012, p. 19-24. ISSN 2232-32366
- PALACIO, M. P. – SOL, D. – GONZÁLEZ, J. 2003.** *Graph-based knowledge representation for GIS data*. In Computer Science, ISSN 0-7695-1915-6, 2003, p. 117-124.
- PATRA, P. 2013.** *Remote Sensing and Geographical Information System (GIS)*. [online]. The Association for Geographical Studies. Delhi, University of Delhi. 28 p. [cit. 5.8.2014]. Dostupné na internete:

http://ags.geography.du.ac.in/Study%20Materials_files/Punyatoya%20Patra_AM.pdf

PODHORÁNYI, M. et al. 2011. *Digitálne výškové dáta pre modelovanie povodní: hodnotenie presnosti DMR.* In Sborník - Symposium GIS Ostrava 2011. Ostrava : VŠB - Technická univerzita Ostrava, 2011. ISBN 978-80-248-2366-9.

RAMACHANDRAN, S. 2010. *Application of Remote Sensing and GIS.* In Current Science, 2010, vol. 75(3) p. 236-244.

RAPANT, P. 1997. *Problémy spojené se zavedením času do databází GISů.* Seč : Referát prednesený na medzinárodnej konferencii „GIS v zemědělství a lesnictví 97“, 1997.

RAPANT, P. 2002. *Úvod do geografických informačních systémů.* Ostrava : VŠB – TU Ostrava, 2002. 112 s.

RAPANT, P. 2005. *Geoinformační technologie.* 2. vyd. Ostrava : Inštitút geoinformatiky, VŠB – TU, 2005. 125 s. ISBN 80-248-1263-0

SANTOS, Y. M. – MOREIRA, A. 2009. *How a Circular Spatially Extended Point is Topological Related with a Line?.* In Proceedings of the World Congress on Engineering. Vol. 1. London : Newswood Limited, 2009. ISBN 978-988-17012-5-1

SHERMAN, G. et al. 2007. *Quantum GIS User Guide.* 2007. 91 s.

SMERNICA EURÓPSKEHO PARLAMENTU A RADY Č. 2007/60/ES, 2007. *o hodnotení a manažmente povodňových rizík.*

STALLMAN, R. 2002. *Free Software, Free Society: Selected Essays of Richard M. Stallman.* Boston : Free Software Foundation, 2002. 230 s. ISBN 1-882114-98-1.

STEYAERT, L. – GOODCHILD, M. F. 1993. *Integrating geographic information systems and environmental simulation models: a status review.* In Environmental information management and analysis: ecosystem to global scales, 1993, p. 333-356. ISBN-10: 0748401237

ŠTĚDRŇ, B. 2010. *Ochrana a licencování počítačového programu.* Praha : Wolters Kluwer ČR, 2010. 220 s. ISBN 978-80-7357-555-7.

- TABIK, S. – ZAPATA, E. L. – ROMERO, L. F. 2012.** *Simultaneous computation of total viewshed on large high resolution grids.* In International Journal of Geographical Information Science, 27(4), 2012, p. 804-814. ISSN 1362-3087
- THEOBALD, M. D. 2007.** *GIS Concepts and ArcGIS Methods.* 4. vyd. Colorado : Conservation Planning Technologies, 2007. 45 s. ISBN 0-9679208-4-1.
- TOMLIN, D. C. 2012.** *GIS and Cartographic Modeling.* 5. vyd. New York : Esri Press, 2012. 250 p. ISBN 978-1589483095.
- TRIZNA, M. 2010.** *Historické povodne.* [online]. Bratislava, Prírodovedecká fakulta, Univerzita Komenského v Bratislave. 6 s. [cit. 2.2.2015].
Dostupné na internete: <http://www.fyzickageografia.sk/geovedy/texty/trizna.pdf>
- TUČEK, J. 1998.** *Geografické informačné systémy.* Zvolen : TU LF Zvolen, 1998. 86 s
- TUČEK, J. 2002.** Prvotné zdroje údajov pre GIS (aj na Slovensku). [on-line]. In zborník referátov, Vojenský informačný systém o území a prax. 2002, s. 261-282 [cit. 15.7.2014].
Dostupné na internete: http://www.topu.mil.sk/data/att/15595_subor.pdf
- U.S. DEPARTMENT OF TRANSPORTATION. 2001.** *Data Integration Glossary.* [online]. Washington, Federal Highway Administration, Office of Asset Management. 18 p. [cit. 12.9.2014]. Dostupné na internete: <http://isddc.dot.gov/OLPFiles/FHWA/010394.pdf>
- VOŽENÍLEK, V. 1998.** *Geografické informační systémy I.: pojetí, historie, základní komponenty.* Olomouc : Univerzita Palackého Olomouc, 1998. 173 s. ISBN 80-7067-802-X.
- WICKERT, A. 2012.** *GRASS GIS for Geomorphologists: An Introductory Guide.* Colorado : Department of Geological Sciences and Institute of Arctic and Alpine Research University od Colorado, 2012. 67 s.
- ZORÁD, L. 2010.** *Simulovanie záplavových zón na rieke Žitava a Hostianskom potoku.* In Mladí vedci 2010 : Zborník vedeckých prác doktorandov a mladých vedeckých pracovníkov. Nitra : Fakulta prírodných vied Univerzity Konštantína Filozofa v Nitre, 2010. ISBN 978-80-8094-742-2, s. 411-419.

Internetové zdroje:

DAHLKE, K. - ROSE, K. 2015 - <http://book.mathreference.com/prolog>, 2015

GRASS DEVELOPMENT TEAM. 2015 - <http://grass.osgeo.org/grass71/manuals/index.html>, 2015

HORÁK, J. 2002 - <http://gis.vsb.cz/pad/index.htm>, 2015

LAJČIN, T. 2013 - <http://www.ictmanazer.cz/2013/03/open-source-software-v-podniku-vyhody-a-rizika/>, 2015

NARTISS, M. 2012 - <http://grass.osgeo.org/grass64/manuals/r.lake.html>, 2015

PRÍLOHY